

# Contents

## [Virtual Machines Documentation](#)

### [Overview](#)

### [Quickstarts](#)

#### [Create a Linux VM](#)

[CLI](#)

[Portal](#)

[PowerShell](#)

[Terraform](#)

[Bicep](#)

[ARM template](#)

#### [Create a Windows VM](#)

[CLI](#)

[Portal](#)

[PowerShell](#)

[Bicep](#)

[ARM template](#)

### [Tutorials](#)

#### [Linux](#)

[1 - Create / manage VMs](#)

[2 - Create / manage disks](#)

[3 - Automate configuration](#)

[4 - Create VM images](#)

[5 - Highly available VMs](#)

[6 - Create a scale set](#)

[7 - Load balance VMs](#)

[8 - Manage networking](#)

#### [Windows](#)

[1 - Create / manage a VM](#)

[2 - Create / manage disks](#)

[3 - Automate configuration](#)

[4 - Create VM images](#)

[5 - Highly available VMs](#)

[6 - Create a scale set](#)

[7 - Load balance VMs](#)

[8 - Manage networking](#)

## Develop

[REST API](#)

[Python](#)

[Python SDK](#)

[Samples](#)

[.NET](#)

[.NET SDK](#)

[Reference](#)

[Java](#)

[Java SDK](#)

[Reference](#)

[Go](#)

[Go SDK](#)

[Reference](#)

[Azure CLI samples repo](#)

[PowerShell samples repo](#)

[Azure Resource Graph queries](#)

## Workloads

[Red Hat](#)

[Cloud Foundry](#)

[OpenShift](#)

[OpenShift overview](#)

[OpenShift Container Platform 4.x](#)

[OpenShift Container Platform 3.11 prerequisites](#)

[OpenShift Container Platform 3.11](#)

[OpenShift Container Platform 3.11 Marketplace Self-Managed](#)

- [Azure Stack](#)
- [OpenShift Container Platform 3.11 post-deployment tasks](#)
- [Troubleshooting OpenShift Container Platform 3.11 deployments](#)
- [Deploy OKD](#)
- [SAP on Azure](#)
- [Oracle](#)
- [Azure for Gaming](#)
- [Elasticsearch](#)
- [Mainframe rehosting](#)
- [Azure Confidential Compute](#)
- [SQL on Virtual Machines](#)
- [High Performance Computing \(HPC\)](#)
- [Instances](#)
  - [Sizes](#)
    - [Overview](#)
    - [General purpose](#)
      - [Overview](#)
      - [Av1-series retirement](#)
      - [Av2-series](#)
      - [B-series burstable](#)
      - [DCsv2-series](#)
      - [DCsv3 and DCcsv3-series](#)
      - [Dv2 and DSv2-series](#)
      - [Dv3 and DSv3-series](#)
      - [Dv4 and Dsv4-series](#)
      - [Dav4 and Dasv4-series](#)
      - [Ddv4 and Ddsv4-series](#)
      - [Dv5 and Dsv5-series](#)
      - [Ddv5 and Ddsv5-series](#)
      - [Dasv5 and Dadsv5-series](#)
      - [DCasv5 and DCadsv5-series](#)
      - [Dpsv5 and Dpdsrv5-series](#)

## Dplsv5 and Dpldsv5-series

Compute optimized

Overview

Fsv2-series

FX-series

Memory optimized

Overview

Dv2 and DSv2-series 11-15

Ev3 and Esv3-series

Eav4 and Easv4-series

Edv4 and Edsv4-series

Ev4 and Esv4-series

Ev5 and Esv5-series

Ebdsv5 and Ebsv5 series

Edv5 and Edsv5-series

Easv5 and Eadsv5-series

ECasv5 and ECadsv5-series

Epsv5 and Epdsv5-series

M-series

Msv2 and Mdsv2 Medium Memory series

Mv2-series

Constrained vCPUs

Storage optimized

Overview

Lsv2-series

Lsv3-series

Lasv3-series

Optimize performance

Linux

Windows

GPU - accelerated compute

Overview

- [NC-series](#)
- [NCv2-series](#)
- [NCv3-series](#)
- [NCasT4\\_v3-series](#)
- [NC\\_A100\\_v4-series](#)
- [NDasrA100\\_v4-series](#)
- [NDm\\_A100\\_v4-series](#)
- [ND-series](#)
- [NDv2-series](#)
- [NV-series](#)
- [NVv3-series](#)
- [NVv4-series](#)
- [NVadsA10\\_v5-series](#)
- [Setup NVIDIA GPU drivers](#)
  - [Linux](#)
  - [Windows](#)
- [Setup AMD GPU drivers](#)
- [NC-series retirement](#)
- [NCv2-series retirement](#)
- [ND-series retirement](#)
- [GPU compute migration guide](#)
- [NV-series retirement](#)
- [NV-series migration guide](#)
- [FPGA - accelerated compute](#)
  - [Overview](#)
  - [NP-series](#)
  - [FPGA Attestation Service](#)
- [High performance compute](#)
  - [Overview](#)
  - [H-series](#)
  - [HB-series](#)
  - [HBv2-series](#)

- [HBv3-series](#)
- [HC-series](#)
- [H-series retirement](#)
- [HB-series retirement](#)
- [Previous generations](#)
- [Generation 2 VMs](#)
- [Isolated sizes](#)
- [Azure compute units \(ACU\)](#)
- [Benchmark scores](#)
  - [Linux](#)
  - [Windows](#)
- [vCPU quotas](#)
  - [CLI](#)
  - [PowerShell](#)
- [Virtual machines selector tool](#)
- [Change the VM size](#)
- [States and billing](#)
- [Azure VMs with no temp disk](#)
- [Azure VM sizes naming conventions](#)
- [Azure Compute Gallery](#)
  - [Overview](#)
  - [Create a gallery](#)
  - [Share a gallery](#)
  - [RBAC](#)
  - [Direct share](#)
  - [Community gallery](#)
- [Images](#)
  - [Images in a gallery](#)
  - [Capture a VM in the portal](#)
  - [Create an image](#)
  - [Create a VM](#)
  - [Generalized](#)

[Specialized](#)

[Update image resources](#)

[Resource Manager Templates](#)

[Create an Azure Compute Gallery](#)

[Create an Image Definition in an Azure Compute Gallery](#)

[Create an Image Version in an Azure Compute Gallery](#)

[Troubleshoot shared images](#)

[Using customer-managed keys](#)

[Export an image to a managed disk](#)

[Purchase plan information](#)

[VM Applications](#)

[Overview](#)

[Deploy VM Applications](#)

[Images](#)

[Find Azure Marketplace images](#)

[Use CLI to find images](#)

[Use PowerShell to find images](#)

[Windows client images](#)

[Windows 10 images](#)

[Linux custom images](#)

[Overview](#)

[Linux provisioning](#)

[Endorsed distributions](#)

[Distribution specific requirements](#)

[Generic steps](#)

[CentOS](#)

[Debian](#)

[Flatcar Container Linux](#)

[FreeBSD](#)

[Oracle Linux](#)

[OpenBSD](#)

[Red Hat](#)

SUSE

Ubuntu

Use cloud-init

Cloud-init overview

Deep dive

Troubleshooting

Configure VM hostname

Update packages in a VM

Add a user on a VM

Configure swapfile

Run existing bash script

Create Linux images without a provisioning agent

Disable Linux Agent provisioning

Windows custom images

Prepare a VHD to Upload

Generalizing a VM

Upload a VHD and create a managed image

Capture a managed image

Create a VM from a managed image

Visual Studio

VM Image Builder

Overview

CLI

Linux

Windows

PowerShell

Windows

Azure Virtual Desktop

Security

Security controls by Azure Policy

Use a virtual network

CLI

- [PowerShell](#)
- [Networking options](#)
- [Configure permissions](#)
  - [CLI](#)
  - [PowerShell](#)
- [DevOps task](#)
- [VM Image Builder template reference](#)
- [Build for image galleries](#)
  - [Linux](#)
  - [Windows](#)
- [Update an existing image](#)
  - [Linux](#)
  - [Windows](#)
- [Store scripts](#)
- [What's new in Azure VM Image Builder](#)
- [Troubleshoot](#)
- [Build image with Packer](#)
  - [Linux](#)
  - [Windows](#)
- [Dedicated hosts](#)
  - [Overview](#)
  - [How-to](#)
    - [General Purpose SKUs](#)
    - [Compute Optimized SKUs](#)
    - [Memory Optimized SKUs](#)
    - [Storage Optimized SKUs](#)
    - [GPU Optimized SKUs](#)
  - [Dedicated Host SKU Retirement](#)
  - [Dedicated Host SKU Migration](#)
- [Azure Spot Virtual Machines](#)
  - [Overview](#)
  - [CLI](#)

- Portal
- PowerShell
- ARM template
- VM size recommendation
- Error codes

- Handle Spot Evictions

## Azure Hybrid Benefit

- Linux
- Linux BYOS
- Windows

## Reserved instances

- What are Azure reservations?
- Prepay for VMs
- Prepay for Dedicated Hosts
- VM instance size flexibility

## Capacity reservation

- Overview
- Create a capacity reservation
- Overallocating capacity reservation
- Modify a capacity reservation
- Associate a VM
- Remove a VM
- Associate a scale set - Flexible
- Associate a scale set - Uniform
- Remove a scale set

## Create Virtual Machines

- Linux
- CLI
- ARM template
- REST
- LAMP stack
- Continuous delivery

- Configure Rolling deployment strategy
- Configure Canary deployment strategy
- Configure Blue-Green deployment strategy
- CI/CD with Azure Pipelines(YAML)

- Secure web server with TLS\SSL

- Windows

- Specialized disk - Portal
- Specialized disk - PowerShell
- ARM template
- Secure web server with TLS\SSL

- Delete a VM and its resources

- Connect to Virtual Machines

- Linux

- Connect to a Linux VM
- Create and manage SSH keys locally
- Create and manage SSH keys in the portal
- Create and manage SSH keys with the Azure CLI
- SSH on Linux or macOS
- SSH on Windows
- Remote Desktop for Linux

- Windows

- Remote Desktop
- SSH
- WinRM

- Time sync

- Linux

- Windows

- Active Directory Windows Virtual Machines in Azure with External NTP Source

- Run Command

- Overview
- Action Run Commands for Linux
- Action Run Commands for Windows

[Managed Run Commands for Linux \(preview\)](#)

[Managed Run Commands for Windows \(preview\)](#)

[Extensions](#)

[Overview](#)

[Linux](#)

[Linux features](#)

[Linux VM Agent](#)

[Windows](#)

[Windows features](#)

[Windows VM Agent](#)

[Azure Backup for SQL Server](#)

[Azure Backup for SQL Server](#)

[Azure Disk Encryption](#)

[Linux](#)

[Windows](#)

[Azure Key Vault](#)

[Linux](#)

[Windows](#)

[Azure Policy guest configuration](#)

[Custom Script](#)

[Linux - version 2](#)

[Windows](#)

[IaaS antimalware for Windows](#)

[VM snapshot](#)

[Linux](#)

[Windows](#)

[Network Watcher Extension](#)

[Linux](#)

[Windows](#)

[Update to latest version](#)

[InfiniBand Drivers](#)

[Linux](#)

[Windows](#)

[NVIDIA GPU Drivers](#)

[Linux](#)

[Windows](#)

[AMD GPU Drivers](#)

[Windows](#)

[Azure Monitor](#)

[Azure Monitor agent](#)

[Diagnostics extension](#)

[Log Analytics](#)

[Linux](#)

[Windows](#)

[VM insights](#)

[Linux Dependency agent](#)

[Windows Dependency agent](#)

[PowerShell DSC](#)

[DSC and Linux](#)

[DSC and Windows](#)

[Handle credentials](#)

[Use templates](#)

[Virtual machine access](#)

[Third-party extensions](#)

[Chef](#)

[Stackify Retrace](#)

[Symantec](#)

[Restrict extension installation](#)

[Linux](#)

[Windows](#)

[Update Linux agent](#)

[Export extensions](#)

[General troubleshooting steps](#)

[Issues with Python 3-enabled Linux systems](#)

[Nested virtualization](#)

[Migrate to Azure Resource Manager](#)

[Retirement starting March 1, 2023](#)

[Migration Overview](#)

[Deep dive on migration](#)

[Plan for migration](#)

[Migrate using the CLI](#)

[Migrate using PowerShell](#)

[Common migration errors](#)

[Community tools for migration](#)

[FAQ](#)

[Availability and scale](#)

[Overview](#)

[Availability zones](#)

[Overview](#)

[CLI](#)

[PowerShell](#)

[Portal](#)

[Migrate to Availability Zones](#)

[Virtual machine scale sets](#)

[Proximity Placement Groups](#)

[Overview](#)

[CLI](#)

[PowerShell](#)

[Portal](#)

[Understand VM reboots](#)

[Azure Regions](#)

[Availability sets](#)

[Overview](#)

[CLI](#)

[PowerShell](#)

[Change an availability set](#)

## Disks

[Overview](#)

[Disk types](#)

[Disk redundancy options](#)

[Deploy a ZRS disk](#)

[Shared disks](#)

[Enable shared disks](#)

[Disk pools](#)

[Overview](#)

[Plan for disk pools](#)

[Deploy a disk pool](#)

[Move a disk pool to a different subscription](#)

[Manage a disk pool](#)

[Deprovision a disk pool](#)

[Troubleshoot a disk pool](#)

[Encryption](#)

[Disk encryption overview](#)

[Server-side encryption](#)

[Server-side encryption overview](#)

[Enable customer-managed keys](#)

[Portal](#)

[PowerShell](#)

[CLI](#)

[Enable cross-tenant customer-managed keys](#)

[Enable encryption at host](#)

[Portal](#)

[PowerShell](#)

[CLI](#)

[Enable double encryption at rest](#)

[Portal](#)

[PowerShell](#)

[CLI](#)

## Azure Disk Encryption

### Linux

[Overview](#)

[Quickstarts](#)

[Azure CLI](#)

[Azure PowerShell](#)

[Azure portal](#)

[Disk encryption scenarios for Linux](#)

[Key vault for Azure Disk Encryption](#)

[Disk encryption sample scripts](#)

[Disk encryption on an isolated network](#)

[How to verify encryption status](#)

[How to configure LVM RAID on crypt](#)

[How to resize encrypted lvm volumes](#)

[Disk encryption troubleshooting](#)

[Disk encryption FAQ](#)

[Upgrade from previous to current version](#)

[Disk encryption - previous version](#)

[Overview](#)

[Key vault for Azure Disk Encryption](#)

[Disk encryption scenarios for Linux](#)

### Windows

[Overview](#)

[Quickstarts](#)

[Azure CLI](#)

[Azure PowerShell](#)

[Azure portal](#)

[Disk encryption scenarios for Windows](#)

[Key vault for Azure Disk Encryption](#)

[Disk encryption sample scripts](#)

[Disk encryption troubleshooting](#)

[Disk encryption FAQ](#)

[Disk encryption - previous version](#)

[Overview](#)

[Key vault for Azure Disk Encryption](#)

[Disk encryption scenarios for Windows](#)

[Performance and cost optimization](#)

[Deploy an ultra disk](#)

[Deploy a premium SSD v2](#)

[Virtual machine and disk performance](#)

[Disk Storage reservations](#)

[Reserve Disk Storage](#)

[Design for high performance](#)

[Disk related metrics](#)

[Disk bursting models](#)

[Enable on-demand bursting](#)

[Disk performance tiers](#)

[Change disk performance tier](#)

[CLI and PowerShell](#)

[Portal](#)

[Enable write accelerator](#)

[Benchmark a disk](#)

[Scalability targets for disks](#)

[Backup and data protection](#)

[Enable incremental snapshots](#)

[Copy incremental snapshots across regions](#)

[Azure Disk Backup](#)

[Overview](#)

[Configure Azure Disk Backup](#)

[Restore disks with Azure Disk Backup](#)

[Backup and disaster recovery for managed disks](#)

[Snapshot a disk](#)

[Backup unmanaged disks](#)

[Backup and disaster recovery for unmanaged disks](#)

## Ephemeral OS disks

[Overview](#)

[Create a VM using ephemeral OS disks](#)

[FAQ on ephemeral OS disks](#)

[Securely import/export a disk](#)

[Configure private links for disks - CLI](#)

[Configure private links for disks - Portal](#)

[Upload a vhd to a disk - PowerShell](#)

[Upload a vhd to a disk - CLI](#)

[Download a VHD](#)

[Linux](#)

[Windows](#)

[Migration and conversion](#)

[Convert disk to another disk type](#)

[CLI](#)

[PowerShell](#)

[Migrate to premium SSDs with Azure Site Recovery](#)

[Linux](#)

[Windows](#)

[Migrate to Managed Disks](#)

[Unmanaged VM to Managed Disks](#)

[CLI](#)

[PowerShell](#)

[Add a data disk](#)

[Linux](#)

[Azure CLI](#)

[Azure portal](#)

[Windows](#)

[Azure PowerShell](#)

[Azure portal](#)

[Change temp disk drive letter](#)

[Detach a disk](#)

[Linux](#)

[Windows](#)

[Expand a disk](#)

[Linux](#)

[Windows](#)

[Unmanaged disks](#)

[Manage storage](#)

[Deploy disks with ARM template](#)

[Use Storage Explorer to manage disks](#)

[Swap the OS disk - CLI](#)

[Swap the OS disk - PowerShell](#)

[Map managed disk to guest disk - Linux](#)

[Map managed disk to guest disk - Windows](#)

[File storage](#)

[Copy files to a VM](#)

[Find unattached disks](#)

[CLI](#)

[PowerShell](#)

[Portal](#)

[Disks FAQs](#)

[Code samples](#)

[CLI](#)

[Create managed disk from a VHD](#)

[Create a managed disk from a snapshot](#)

[Copy a managed disk to the same or different subscription](#)

[Export a snapshot as a VHD to a storage account](#)

[Export a managed disk as a VHD to a storage account](#)

[Copy a snapshot to the same or different subscription](#)

[Create virtual machine from snapshot](#)

[Create virtual machine from existing managed OS disk](#)

[PowerShell](#)

[Create managed disk from a VHD](#)

- Create a managed disk from a snapshot
- Copy a snapshot to the same or different subscription
- Export a snapshot as a VHD to a storage account
- Export a managed disk as a VHD to a storage account
- Create a snapshot from a VHD

## Networking

- Overview

- Optimize network throughput

- Best practices

- Create virtual network

- Filter network traffic

- Azure portal

- Azure PowerShell

- Azure CLI

- Create VM - static public IP

- Azure portal

- Azure PowerShell

- Azure CLI

- Add public IP address to existing VM

- Dissociate public IP address from a VM

- Create VM - static private IP

- Azure portal

- Azure PowerShell

- Azure CLI

- Create VM - multiple IPs

- Azure portal

- Azure PowerShell

- Azure CLI

- Add or remove network interfaces

- Create VM - multiple NICs

- Azure PowerShell

- Azure CLI

Create VM - accelerated networking

Azure PowerShell

Azure CLI

Set up DPDK

TCP/IP performance tuning for Azure VMs

Virtual machine network throughput

Create a DDOS protection plan

Portal

PowerShell

CLI

ARM template

Open ports to a VM

CLI

PowerShell

Portal

Migrate a virtual machine public IP address

Assign public IP address

Use a custom domain name

Use multiple NICs

Linux using CLI

Windows using PowerShell

Assign public DNS name

Assign public DNS name

DNS resolution

Use internal DNS

Security

Overview

Microsoft Defender for Cloud

Security baselines

Linux

Windows

Recommendations

[Trusted launch](#)

[Overview](#)

[Deploy Trusted Launch VM](#)

[Just-in-time access](#)

[Policy](#)

[Overview](#)

[Controls by policy](#)

[Policy reference](#)

[Use policies](#)

[Linux](#)

[Windows](#)

[Bastion](#)

[Overview](#)

[Create a Bastion host](#)

[Portal](#)

[PowerShell](#)

[CLI](#)

[Bastion SSH connection](#)

[Bastion RDP connection](#)

[Use access controls](#)

[Create a Key Vault](#)

[CLI](#)

[PowerShell](#)

[Mitigating speculative execution](#)

[Join a Linux VM to Azure Active Directory](#)

[Red Hat Enterprise Linux](#)

[CentOS](#)

[Ubuntu](#)

[Configure managed identities](#)

[Portal](#)

[CLI](#)

[PowerShell](#)

[Azure Resource Manager Template](#)

[REST](#)

[Azure SDKs](#)

[Linux VMs SSH with Azure AD](#)

[Windows VMs and Azure AD](#)

[Updates and maintenance](#)

[Overview](#)

[Automatic OS image upgrade](#)

[Overview](#)

[Maintenance control for OS image upgrade](#)

[Overview](#)

[PowerShell](#)

[CLI](#)

[Portal](#)

[ARM Template](#)

[Automatic VM guest patching](#)

[Overview](#)

[Hotpatch](#)

[Automatic extension upgrade](#)

[Update Management in Azure Automation](#)

[Maintenance](#)

[Overview](#)

[Maintenance notifications](#)

[Overview](#)

[CLI](#)

[Portal](#)

[PowerShell](#)

[Maintenance configurations](#)

[Overview](#)

[CLI](#)

[PowerShell](#)

[Portal](#)

## Scheduled events

Linux

Windows

Scheduled events

Monitor scheduled events

## Track and update VMs

Linux

Windows

## Monitoring

Monitor virtual machines

Monitor virtual machine reference

## Tutorials

Enable monitoring

Alert on machine down

Collect guest logs and metrics

## Agents

Agents overview

FAQ

### Azure Monitor Agent

Overview

Install and manage

Data collection

Migrate from legacy agents

FAQ

### Log Analytics agent

Overview

Linux agents

Log Analytics gateway

Agent management

Agent Health

Data sources

Overview

- [Custom JSON data](#)
- [collect performance data](#)
- [Syslog](#)
- [Performance counters](#)
- [Linux application performance](#)
- [Custom logs](#)
- [Custom fields](#)
- [Troubleshoot](#)
  - [Log Analytics VM Extension](#)
  - [Log Analytics Linux agent](#)
- [VM insights](#)
  - [Overview](#)
  - [FAQ](#)
  - [Enable monitoring](#)
    - [Enable monitoring overview](#)
    - [Enable for single Azure VM](#)
    - [Enable using Azure Policy](#)
    - [Enable using Azure PowerShell](#)
    - [Enable for Hybrid environment](#)
  - [Map dependencies](#)
  - [Monitor performance](#)
    - [Analyze data with log queries](#)
    - [Visualize data with workbooks](#)
    - [Create alert rules](#)
    - [Upgrade Dependency agent](#)
    - [Disable monitoring](#)
  - [VM usage](#)
  - [Tag a VM](#)
    - [CLI](#)
    - [Portal](#)
    - [PowerShell](#)
    - [Template](#)

[Monitor metadata](#)

[CLI](#)

[PowerShell](#)

[Get usage metrics with REST](#)

[Boot diagnostics](#)

[Backup and recovery](#)

[Overview](#)

[Service disruptions](#)

[Back up VMs](#)

[Overview](#)

[Quickstarts](#)

[CLI](#)

[PowerShell](#)

[Portal](#)

[Template](#)

[Back up multiple VMs](#)

[Restore options](#)

[Support matrix](#)

[Pricing](#)

[FAQ](#)

[Disaster recovery](#)

[Overview](#)

[Enable disaster recovery](#)

[Linux](#)

[Windows](#)

[Fail over a VM to another region](#)

[Fail back a VM to the primary region](#)

[VM restore points](#)

[Overview](#)

[Quickstarts](#)

[Create VM restore points](#)

[Manage VM restore points](#)

## Tutorials

[CLI](#)

[PowerShell](#)

[Portal](#)

[Support matrix](#)

[Troubleshooting](#)

## Move and migrate VMs

[Change subscription or resource group](#)

[CLI](#)

[PowerShell](#)

[Change subscription for Marketplace VMs](#)

[Move VMs to another region](#)

[Move to an availability zone](#)

[Move Maintenance Control configurations to another region](#)

[Move Maintenance Control configuration resources to another region](#)

[Migrate AWS and on-premises VMs](#)

[Migrate from Amazon Web Services \(AWS\) to Azure](#)

[Upload on-premises VM](#)

[Use Azure Site Recovery](#)

## Infrastructure automation

[Overview](#)

[User data](#)

[Custom data](#)

[Ansible](#)

[Terraform](#)

[Jenkins](#)

[Azure DevOps](#)

## Resources

[Cloud adoption framework](#)

[Architecture center](#)

[Migration tools](#)

[Microsoft Q&A](#)

[Azure Quickstart Templates](#)

[Pricing](#)

[Regional availability](#)

[Build your skills with Microsoft Learn training](#)

[Azure Roadmap](#)

[Pricing calculator](#)

[Common CLI commands](#)

[Common PowerShell commands](#)

[Common networking PowerShell commands](#)

[VM template description](#)

[Classic deployments](#)

[Linux VMs using classic deployment](#)

[Windows VMs using classic deployment](#)

[FAQ](#)

[Linux](#)

[Windows](#)

[Support and troubleshooting](#)



# Virtual machines in Azure

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

Azure virtual machines are one of several types of [on-demand, scalable computing resources](#) that Azure offers. Typically, you choose a virtual machine when you need more control over the computing environment than the other choices offer. This article gives you information about what you should consider before you create a virtual machine, how you create it, and how you manage it.

An Azure virtual machine gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it. However, you still need to maintain the virtual machine by performing tasks, such as configuring, patching, and installing the software that runs on it.

Azure virtual machines can be used in various ways. Some examples are:

- **Development and test** – Azure virtual machines offer a quick and easy way to create a computer with specific configurations required to code and test an application.
- **Applications in the cloud** – Because demand for your application can fluctuate, it might make economic sense to run it on a virtual machine in Azure. You pay for extra virtual machines when you need them and shut them down when you don't.
- **Extended datacenter** – virtual machines in an Azure virtual network can easily be connected to your organization's network.

The number of virtual machines that your application uses can scale up and out to whatever is required to meet your needs.

## What do I need to think about before creating a virtual machine?

There is always a multitude of [design considerations](#) when you build out an application infrastructure in Azure. These aspects of a virtual machine are important to think about before you start:

- The names of your application resources
- The location where the resources are stored
- The size of the virtual machine
- The maximum number of virtual machines that can be created
- The operating system that the virtual machine runs
- The configuration of the virtual machine after it starts
- The related resources that the virtual machine needs

### Locations

There are multiple [geographical regions](#) around the world where you can create Azure resources. Usually, the region is called **location** when you create a virtual machine. For a virtual machine, the location specifies where the virtual hard disks will be stored.

This table shows some of the ways you can get a list of available locations.

METHOD	DESCRIPTION
Azure portal	Select a location from the list when you create a virtual machine.

METHOD	DESCRIPTION
Azure PowerShell	Use the <a href="#">Get-AzLocation</a> command.
REST API	Use the <a href="#">List locations</a> operation.
Azure CLI	Use the <a href="#">az account list-locations</a> operation.

## Availability

There are multiple options to manage the availability of your virtual machines in Azure.

- **Availability Zones** are physically separated zones within an Azure region. Availability zones guarantee you will have virtual machine Connectivity to at least one instance at least 99.99% of the time when you have two or more instances deployed across two or more Availability Zones in the same Azure region.
- **Virtual machine scale sets** let you create and manage a group of load balanced virtual machines. The number of virtual machine instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update many virtual machines. Virtual machines in a scale set can also be deployed into multiple availability zones, a single availability zone, or regionally.
- **Proximity Placement Groups** are a grouping construct used to ensure Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

For more information see [Availability options for Azure virtual machines](#) and [SLA for Azure virtual machines](#).

## Virtual machine size

The [size](#) of the virtual machine that you use is determined by the workload that you want to run. The size that you choose then determines factors such as processing power, memory, storage capacity, and network bandwidth. Azure offers a wide variety of sizes to support many types of uses.

Azure charges an [hourly price](#) based on the virtual machine's size and operating system. For partial hours, Azure charges only for the minutes used. Storage is priced and charged separately.

## Virtual machine limits

Your subscription has default [quota limits](#) in place that could impact the deployment of many virtual machines for your project. The current limit on a per subscription basis is 20 virtual machines per region. Limits can be raised by [filing a support ticket requesting an increase](#)

## Managed Disks

Managed Disks handles Azure Storage account creation and management in the background for you, and ensures that you do not have to worry about the scalability limits of the storage account. You specify the disk size and the performance tier (Standard or Premium), and Azure creates and manages the disk. As you add disks or scale the virtual machine up and down, you don't have to worry about the storage being used. If you're creating new virtual machines, [use the Azure CLI](#) or the Azure portal to create virtual machines with Managed OS and data disks. If you have virtual machines with unmanaged disks, you can [convert your virtual machines to be backed with Managed Disks](#).

You can also manage your custom images in one storage account per Azure region, and use them to create hundreds of virtual machines in the same subscription. For more information about Managed Disks, see the

## Distributions

Microsoft Azure supports a variety of Linux and Windows distributions. You can find available distributions in the [marketplace](#), Azure portal or by querying results using CLI, PowerShell and REST APIs.

This table shows some ways that you can find the information for an image.

METHOD	DESCRIPTION
Azure portal	The values are automatically specified for you when you select an image to use.
Azure PowerShell	<code>Get-AzVMImagePublisher</code> -Location <i>location</i> <code>Get-AzVMImageOffer</code> -Location <i>location</i> -Publisher <i>publisherName</i> <code>Get-AzVMImageSku</code> -Location <i>location</i> -Publisher <i>publisherName</i> -Offer <i>offerName</i>
REST APIs	<a href="#">List image publishers</a> <a href="#">List image offers</a> <a href="#">List image skus</a>
Azure CLI	<code>az vm image list-publishers --location <i>location</i></code> <code>az vm image list-offers --location <i>location</i> --publisher <i>publisherName</i></code> <code>az vm image list-skus --location <i>location</i> --publisher <i>publisherName</i> --offer <i>offerName</i></code>

Microsoft works closely with partners to ensure the images available are updated and optimized for an Azure runtime. For more information on Azure partner offers, see the following links:

- Linux on Azure - [Endorsed Distributions](#)
- SUSE - [Azure Marketplace](#) - SUSE Linux Enterprise Server
- Red Hat - [Azure Marketplace](#) - Red Hat Enterprise Linux
- Canonical - [Azure Marketplace](#) - Ubuntu Server
- Debian - [Azure Marketplace](#) - Debian
- FreeBSD - [Azure Marketplace](#) - FreeBSD
- Flatcar - [Azure Marketplace](#) - Flatcar Container Linux
- RancherOS - [Azure Marketplace](#) - RancherOS
- Bitnami - [Bitnami Library for Azure](#)
- Mesosphere - [Azure Marketplace](#) - Mesosphere DC/OS on Azure
- Docker - [Azure Marketplace](#) - Docker images
- Jenkins - [Azure Marketplace](#) - CloudBees Jenkins Platform

## Cloud-init

To achieve a proper DevOps culture, all infrastructures must be code. When all the infrastructure lives in code it can easily be recreated. Azure works with all the major automation tooling like Ansible, Chef, SaltStack, and Puppet. Azure also has its own tooling for automation:

- [Azure Templates](#)
- [Azure VMaccess](#)

Azure supports for [cloud-init](#) across most Linux Distros that support it. We are actively working with our endorsed Linux distro partners in order to have cloud-init enabled images available in the Azure marketplace. These images will make your cloud-init deployments and configurations work seamlessly with virtual machines and virtual machine scale sets.

- [Using cloud-init on Azure Linux virtual machines](#)

## Storage

- [Introduction to Microsoft Azure Storage](#)
- [Add a disk to a Linux virtual machine using the azure-cli](#)
- [How to attach a data disk to a Linux virtual machine in the Azure portal](#)

## Networking

- [Virtual Network Overview](#)
- [IP addresses in Azure](#)
- [Opening ports to a Linux virtual machine in Azure](#)
- [Create a Fully Qualified Domain Name in the Azure portal](#)

## Data residency

In Azure, the feature to enable storing customer data in a single region is currently only available in the Southeast Asia Region (Singapore) of the Asia Pacific Geo and Brazil South (Sao Paulo State) Region of Brazil Geo. For all other regions, customer data is stored in Geo. For more information, see [Trust Center](#).

## Next steps

Create your first virtual machine!

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)

# Quickstart: Create a Linux virtual machine with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This quickstart shows you how to use the Azure CLI to deploy a Linux virtual machine (VM) in Azure. The Azure CLI is used to create and manage Azure resources via either the command line or scripts.

In this tutorial, we will be installing the latest Debian image. To show the VM in action, you'll connect to it using SSH and install the NGINX web server.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also open Cloud Shell in a separate browser tab by going to <https://shell.azure.com/bash>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and select **Enter** to run it.

If you prefer to install and use the CLI locally, this quickstart requires Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create a resource group

Create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name myResourceGroup --location eastus
```

## Create virtual machine

Create a VM with the [az vm create](#) command.

The following example creates a VM named *myVM* and adds a user account named *azureuser*. The `--generate-ssh-keys` parameter is used to automatically generate an SSH key, and put it in the default key location (`~/.ssh`). To use a specific set of keys instead, use the `--ssh-key-values` option.

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image Debian \
--admin-username azureuser \
--generate-ssh-keys
```

It takes a few minutes to create the VM and supporting resources. The following example output shows the VM create operation was successful.

```
{  
    "fqdns": "",  
    "id":  
        "/subscriptions/<guid>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",  
    "location": "eastus",  
    "macAddress": "00-0D-3A-23-9A-49",  
    "powerState": "VM running",  
    "privateIpAddress": "10.0.0.4",  
    "publicIpAddress": "40.68.254.142",  
    "resourceGroup": "myResourceGroup"  
}
```

Make a note of the `publicIpAddress` to use later.

## Install web server

To see your VM in action, install the NGINX web server. Update your package sources and then install the latest NGINX package.

```
az vm run-command invoke \  
-g myResourceGroup \  
-n myVM \  
--command-id RunShellScript \  
--scripts "sudo apt-get update && sudo apt-get install -y nginx"
```

## Open port 80 for web traffic

By default, only SSH connections are opened when you create a Linux VM in Azure. Use `az vm open-port` to open TCP port 80 for use with the NGINX web server:

```
az vm open-port --port 80 --resource-group myResourceGroup --name myVM
```

## View the web server in action

Use a web browser of your choice to view the default NGINX welcome page. Use the public IP address of your VM as the web address. The following example shows the default NGINX web site:



## Clean up resources

When no longer needed, you can use the `az group delete` command to remove the resource group, VM, and all

related resources.

```
az group delete --name myResourceGroup
```

## Next steps

In this quickstart, you deployed a simple virtual machine, opened a network port for web traffic, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Create a Linux virtual machine in the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Azure virtual machines (VMs) can be created through the Azure portal. The Azure portal is a browser-based user interface to create Azure resources. This quickstart shows you how to use the Azure portal to deploy a Linux virtual machine (VM) running Ubuntu 18.04 LTS. To see your VM in action, you also SSH to the VM and install the NGINX web server.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Sign in to Azure

Sign in to the [Azure portal](#).

## Create virtual machine

1. Enter *virtual machines* in the search.
2. Under **Services**, select **Virtual machines**.
3. In the **Virtual machines** page, select **Create** and then **Virtual machine**. The **Create a virtual machine** page opens.
4. In the **Basics** tab, under **Project details**, make sure the correct subscription is selected and then choose to **Create new resource group**. Enter *myResourceGroup* for the name.\*.

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

<b>Subscription</b> * ⓘ	<input type="text" value="Pay-As-You-Go"/>
<b>Resource group</b> * ⓘ	<input type="text" value="(New) myResourceGroup"/> <a href="#">Create new</a>

5. Under **Instance details**, enter *myVM* for the **Virtual machine name**, and choose *Ubuntu 18.04 LTS - Gen2* for your **Image**. Leave the other defaults. The default size and pricing is only shown as an example. Size availability and pricing are dependent on your region and subscription.

**Instance details**

Virtual machine name *	myVM
Region *	(US) East US
Availability options	No infrastructure redundancy required
Security type	Standard
Image *	Ubuntu Server 18.04 LTS - Gen2
See all images   Configure VM generation	
Azure Spot instance	<input type="checkbox"/>
Size *	Standard_DS1_v2 - 1 vcpu, 3.5 GiB memory
See all sizes	

#### NOTE

Some users will now see the option to create VMs in multiple zones. To learn more about this new capability, see [Create virtual machines in an availability zone](#).

Availability zone *	Zones 1
You can now select multiple zones. Selecting multiple zones will create one VM per zone.	

6. Under **Administrator account**, select **SSH public key**.
7. In **Username** enter *azureuser*.
8. For **SSH public key source**, leave the default of **Generate new key pair**, and then enter *myKey* for the **Key pair name**.

**Administrator account**

Authentication type	<input checked="" type="radio"/> SSH public key <input type="radio"/> Password
Username *	azureuser
SSH public key source	Generate new key pair
Key pair name *	myKey

9. Under **Inbound port rules > Public inbound ports**, choose **Allow selected ports** and then select **SSH (22)** and **HTTP (80)** from the drop-down.

**Inbound port rules**

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports *	<input type="radio"/> None <input checked="" type="radio"/> Allow selected ports
Select inbound ports *	HTTP (80), SSH (22)
This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.	

10. Leave the remaining defaults and then select the **Review + create** button at the bottom of the page.
11. On the **Create a virtual machine** page, you can see the details about the VM you are about to create.

When you are ready, select **Create**.

12. When the **Generate new key pair** window opens, select **Download private key and create resource**. Your key file will be download as **myKey.pem**. Make sure you know where the **.pem** file was downloaded; you will need the path to it in the next step.
13. When the deployment is finished, select **Go to resource**.
14. On the page for your new VM, select the public IP address and copy it to your clipboard.



## Connect to virtual machine

Create an SSH connection with the VM.

1. If you are on a Mac or Linux machine, open a Bash prompt and set read-only permission on the **.pem** file using `chmod 400 ~/Downloads/myKey.pem`. If you are on a Windows machine, open a PowerShell prompt.
2. At your prompt, open an SSH connection to your virtual machine. Replace the IP address with the one from your VM, and replace the path to the **.pem** with the path to where the key file was downloaded.

```
ssh -i ~/Downloads/myKey.pem azureuser@10.111.12.123
```

### TIP

The SSH key you created can be used the next time you create a VM in Azure. Just select the **Use a key stored in Azure for SSH public key source** the next time you create a VM. You already have the private key on your computer, so you won't need to download anything.

## Install web server

To see your VM in action, install the NGINX web server. From your SSH session, update your package sources and then install the latest NGINX package.

```
sudo apt-get -y update  
sudo apt-get -y install nginx
```

When done, type `exit` to leave the SSH session.

## View the web server in action

Use a web browser of your choice to view the default NGINX welcome page. Type the public IP address of the VM as the web address. The public IP address can be found on the VM overview page or as part of the SSH connection string you used earlier.



## Clean up resources

When no longer needed, you can delete the resource group, virtual machine, and all related resources. To do so, select the resource group for the virtual machine, select **Delete**, then confirm the name of the resource group to delete.

## Next steps

In this quickstart, you deployed a simple virtual machine, created a Network Security Group and rule, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Create a Linux virtual machine in Azure with PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

The Azure PowerShell module is used to create and manage Azure resources from the PowerShell command line or in scripts. This quickstart shows you how to use the Azure PowerShell module to deploy a Linux virtual machine (VM) in Azure. This quickstart uses the Ubuntu 18.04 LTS marketplace image from Canonical. To see your VM in action, you'll also SSH to the VM and install the NGINX web server.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create a resource group

Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which Azure resources are deployed and managed:

```
New-AzResourceGroup -Name 'myResourceGroup' -Location 'EastUS'
```

## Create a virtual machine

We will be automatically generating an SSH key pair to use for connecting to the VM. The public key that is created using `-GenerateSshKey` will be stored in Azure as a resource, using the name you provide as `SshKeyName`. The SSH key resource can be reused for creating additional VMs. Both the public and private keys will also be downloaded for you. When you create your SSH key pair using the Cloud Shell, the keys are stored in a [storage account that is automatically created by Cloud Shell](#). Don't delete the storage account, or the file share in it, until after you have retrieved your keys or you will lose access to the VM.

You will be prompted for a user name that will be used when you connect to the VM. You will also be asked for a password, which you can leave blank. Password login for the VM is disabled when using an SSH key.

In this example, you create a VM named *myVM*, in *East US*, using the *Standard\_B2s* VM size.

```
New-AzVm ` 
    -ResourceGroupName 'myResourceGroup' ` 
    -Name 'myVM' ` 
    -Location 'East US' ` 
    -Image Debian ` 
    -size Standard_B2s ` 
    -PublicIpAddressName myPubIP ` 
    -OpenPorts 80 ` 
    -GenerateSshKey ` 
    -SshKeyName mySSHKey
```

The output will give you the location of the local copy of the SSH key. For example:

```
Private key is saved to /home/user/.ssh/1234567891
Public key is saved to /home/user/.ssh/1234567891.pub
```

It will take a few minutes for your VM to be deployed. When the deployment is finished, move on to the next section.

## Install NGINX

To see your VM in action, install the NGINX web server.

```
Invoke-AzVMRunCommand ` 
    -ResourceGroupName 'myResourceGroup' ` 
    -Name 'myVM' ` 
    -CommandId 'RunShellScript' ` 
    -ScriptString 'sudo apt-get update && sudo apt-get install -y nginx'
```

The `-ScriptString` parameter requires version 4.27.0 or later of the `'Az.Compute'` module.

## View the web server in action

Get the public IP address of your VM:

```
Get-AzPublicIpAddress -Name myPubIP -ResourceGroupName myResourceGroup | select "IpAddress"
```

Use a web browser of your choice to view the default NGINX welcome page. Enter the public IP address of the VM as the web address.



A screenshot of a web browser window. The address bar shows the IP address 40.68.254.142. The main content area displays the text "Welcome to nginx on Debian!". Below it, there is a message: "If you see this page, the nginx web server is successfully installed and working on Debian. Further configuration is required." Further down, it says: "For online documentation and support please refer to [nginx.org](http://nginx.org)". At the bottom, there is a note: "Please use the reportbug tool to report bugs in the nginx package with Debian. However, check [existing bug reports](#) before reporting a new bug." The browser interface includes standard controls like back, forward, and search.

**Welcome to nginx on Debian!**

If you see this page, the nginx web server is successfully installed and working on Debian. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org)

Please use the `reportbug` tool to report bugs in the `nginx` package with Debian. However, check [existing bug reports](#) before reporting a new bug.

*Thank you for using debian and nginx.*

## Clean up resources

When no longer needed, you can use the [Remove-AzResourceGroup](#) cmdlet to remove the resource group, VM, and all related resources:

```
Remove-AzResourceGroup -Name 'myResourceGroup'
```

## Next steps

In this quickstart, you deployed a simple virtual machine, created a Network Security Group and rule, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Use Terraform to create a Linux VM

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Article tested with the following Terraform and Terraform provider versions:

- [Terraform v1.2.7](#)
- [AzureRM Provider v3.20.0](#)

This article shows you how to create a complete Linux environment and supporting resources with Terraform. Those resources include a virtual network, subnet, public IP address, and more.

Terraform enables the definition, preview, and deployment of cloud infrastructure. Using Terraform, you create configuration files using [HCL syntax](#). The HCL syntax allows you to specify the cloud provider - such as Azure - and the elements that make up your cloud infrastructure. After you create your configuration files, you create an *execution plan* that allows you to preview your infrastructure changes before they're deployed. Once you verify the changes, you apply the execution plan to deploy the infrastructure.

In this article, you learn how to:

- Create a virtual network
- Create a subnet
- Create a public IP address
- Create a network security group and SSH inbound rule
- Create a virtual network interface card
- Connect the network security group to the network interface
- Create a storage account for boot diagnostics
- Create SSH key
- Create a virtual machine
- Use SSH to connect to virtual machine

## NOTE

The example code in this article is located in the [Microsoft Terraform GitHub repo](#). See more [articles and sample code showing how to use Terraform to manage Azure resources](#)

## Prerequisites

- **Azure subscription:** If you don't have an Azure subscription, create a [free account](#) before you begin.
- [Install and configure Terraform](#)

## Implement the Terraform code

1. Create a directory in which to test the sample Terraform code and make it the current directory.
2. Create a file named `providers.tf` and insert the following code:

```

terraform {
  required_version = ">=0.12"

  required_providers {
    azurerm = {
      source  = "hashicorp/azurerm"
      version = "~>2.0"
    }
    random = {
      source  = "hashicorp/random"
      version = "~>3.0"
    }
    tls = {
      source  = "hashicorp/tls"
      version = "~>4.0"
    }
  }
}

provider "azurerm" {
  features {}
}

```

3. Create a file named `main.tf` and insert the following code:

```

resource "random_pet" "rg_name" {
  prefix = var.resource_group_name_prefix
}

resource "azurerm_resource_group" "rg" {
  location = var.resource_group_location
  name     = random_pet.rg_name.id
}

# Create virtual network
resource "azurerm_virtual_network" "my_terraform_network" {
  name          = "myVnet"
  address_space = ["10.0.0.0/16"]
  location      = azurerm_resource_group.rg.location
  resource_group_name = azurerm_resource_group.rg.name
}

# Create subnet
resource "azurerm_subnet" "my_terraform_subnet" {
  name          = "mySubnet"
  resource_group_name = azurerm_resource_group.rg.name
  virtual_network_name = azurerm_virtual_network.my_terraform_network.name
  address_prefixes = ["10.0.1.0/24"]
}

# Create public IPs
resource "azurerm_public_ip" "my_terraform_public_ip" {
  name          = "myPublicIP"
  location      = azurerm_resource_group.rg.location
  resource_group_name = azurerm_resource_group.rg.name
  allocation_method = "Dynamic"
}

# Create Network Security Group and rule
resource "azurerm_network_security_group" "my_terraform_nsg" {
  name          = "myNetworkSecurityGroup"
  location      = azurerm_resource_group.rg.location
  resource_group_name = azurerm_resource_group.rg.name

  security_rule {
    name          = "SSH"
  }
}

```

```

        priority          = 1001
        direction        = "Inbound"
        access           = "Allow"
        protocol         = "Tcp"
        source_port_range = "*"
        destination_port_range = "22"
        source_address_prefix = "*"
        destination_address_prefix = "*"
    }
}

# Create network interface
resource "azurerm_network_interface" "my_terraform_nic" {
    name            = "myNIC"
    location        = azurerm_resource_group.rg.location
    resource_group_name = azurerm_resource_group.rg.name

    ip_configuration {
        name          = "my_nic_configuration"
        subnet_id     = azurerm_subnet.my_terraform_subnet.id
        private_ip_address_allocation = "Dynamic"
        public_ip_address_id       = azurerm_public_ip.my_terraform_public_ip.id
    }
}

# Connect the security group to the network interface
resource "azurerm_network_interface_security_group_association" "example" {
    network_interface_id      = azurerm_network_interface.my_terraform_nic.id
    network_security_group_id = azurerm_network_security_group.my_terraform_nsg.id
}

# Generate random text for a unique storage account name
resource "random_id" "random_id" {
    keepers = {
        # Generate a new ID only when a new resource group is defined
        resource_group = azurerm_resource_group.rg.name
    }

    byte_length = 8
}

# Create storage account for boot diagnostics
resource "azurerm_storage_account" "my_storage_account" {
    name          = "diag${random_id.random_id.hex}"
    location      = azurerm_resource_group.rg.location
    resource_group_name = azurerm_resource_group.rg.name
    account_tier      = "Standard"
    account_replication_type = "LRS"
}

# Create (and display) an SSH key
resource "tls_private_key" "example_ssh" {
    algorithm = "RSA"
    rsa_bits   = 4096
}

# Create virtual machine
resource "azurerm_linux_virtual_machine" "my_terraform_vm" {
    name            = "myVM"
    location        = azurerm_resource_group.rg.location
    resource_group_name = azurerm_resource_group.rg.name
    network_interface_ids = [azurerm_network_interface.my_terraform_nic.id]
    size           = "Standard_DS1_v2"

    os_disk {
        name          = "myOsDisk"
        caching        = "ReadWrite"
        storage_account_type = "Premium_LRS"
    }
}

```

```

source_image_reference {
  publisher = "Canonical"
  offer     = "UbuntuServer"
  sku       = "18.04-LTS"
  version   = "latest"
}

computer_name          = "myvm"
admin_username         = "azureuser"
disable_password_authentication = true

admin_ssh_key {
  username  = "azureuser"
  public_key = tls_private_key.example_ssh.public_key_openssh
}

boot_diagnostics {
  storage_account_uri = azurerm_storage_account.my_storage_account.primary_blob_endpoint
}
}

```

4. Create a file named `variables.tf` and insert the following code:

```

variable "resource_group_location" {
  default      = "eastus"
  description = "Location of the resource group."
}

variable "resource_group_name_prefix" {
  default      = "rg"
  description = "Prefix of the resource group name that's combined with a random ID so name is unique
in your Azure subscription."
}

```

5. Create a file named `outputs.tf` and insert the following code:

```

output "resource_group_name" {
  value = azurerm_resource_group.rg.name
}

output "public_ip_address" {
  value = azurerm_linux_virtual_machine.my_terraform_vm.public_ip_address
}

output "tls_private_key" {
  value      = tls_private_key.example_ssh.private_key_pem
  sensitive = true
}

```

## Initialize Terraform

Run `terraform init` to initialize the Terraform deployment. This command downloads the Azure modules required to manage your Azure resources.

```
terraform init
```

## Create a Terraform execution plan

Run [terraform plan](#) to create an execution plan.

```
terraform plan -out main.tfplan
```

#### Key points:

- The `terraform plan` command creates an execution plan, but doesn't execute it. Instead, it determines what actions are necessary to create the configuration specified in your configuration files. This pattern allows you to verify whether the execution plan matches your expectations before making any changes to actual resources.
- The optional `-out` parameter allows you to specify an output file for the plan. Using the `-out` parameter ensures that the plan you reviewed is exactly what is applied.
- To read more about persisting execution plans and security, see the [security warning section](#).

## Apply a Terraform execution plan

Run [terraform apply](#) to apply the execution plan to your cloud infrastructure.

```
terraform apply main.tfplan
```

#### Key points:

- The `terraform apply` command above assumes you previously ran `terraform plan -out main.tfplan`.
- If you specified a different filename for the `-out` parameter, use that same filename in the call to `terraform apply`.
- If you didn't use the `-out` parameter, call `terraform apply` without any parameters.

## Verify the results

To use SSH to connect to the virtual machine, do the following steps:

1. Run [terraform output](#) to get the SSH private key and save it to a file.

```
terraform output -raw tls_private_key > id_rsa
```

2. Run [terraform output](#) to get the virtual machine public IP address.

```
terraform output public_ip_address
```

3. Use SSH to connect to the virtual machine.

```
ssh -i id_rsa azureuser@<public_ip_address>
```

#### Key points:

- Depending on the permissions of your environment, you might get an error when trying to ssh into the virtual machine using the `id_rsa` key file. If you get an error stating that the private key file is unprotected and can't be used, try running the following command: `chmod 600 id_rsa`, which will restrict read and write access to the owner of the file.

## Clean up resources

When you no longer need the resources created via Terraform, do the following steps:

1. Run `terraform plan` and specify the `-destroy` flag.

```
terraform plan -destroy -out main.destroy.tfplan
```

#### Key points:

- The `terraform plan` command creates an execution plan, but doesn't execute it. Instead, it determines what actions are necessary to create the configuration specified in your configuration files. This pattern allows you to verify whether the execution plan matches your expectations before making any changes to actual resources.
- The optional `-out` parameter allows you to specify an output file for the plan. Using the `-out` parameter ensures that the plan you reviewed is exactly what is applied.
- To read more about persisting execution plans and security, see the [security warning section](#).

2. Run `terraform apply` to apply the execution plan.

```
terraform apply main.destroy.tfplan
```

## Troubleshoot Terraform on Azure

[Troubleshoot common problems when using Terraform on Azure](#)

## Next steps

In this quickstart, you deployed a simple virtual machine using Terraform. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Create an Ubuntu Linux virtual machine using a Bicep file

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This quickstart shows you how to use a Bicep file to deploy an Ubuntu Linux virtual machine (VM) in Azure.

**Bicep** is a domain-specific language (DSL) that uses declarative syntax to deploy Azure resources. It provides concise syntax, reliable type safety, and support for code reuse. Bicep offers the best authoring experience for your infrastructure-as-code solutions in Azure.

## Prerequisites

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Review the Bicep file

The Bicep file used in this quickstart is from [Azure Quickstart Templates](#).

```
@description('The name of your Virtual Machine.')
param vmName string = 'simpleLinuxVM'

@description('Username for the Virtual Machine.')
param adminUsername string

@description('Type of authentication to use on the Virtual Machine. SSH key is recommended.')
@allowed([
    'sshPublicKey'
    'password'
])
param authenticationType string = 'password'

@description('SSH Key or password for the Virtual Machine. SSH key is recommended.')
@secure()
param adminPasswordOrKey string

@description('Unique DNS Name for the Public IP used to access the Virtual Machine.')
param dnsLabelPrefix string = toLower('${vmName}-${uniqueString(resourceGroup().id)}')

@description('The Ubuntu version for the VM. This will pick a fully patched image of this given Ubuntu
version.')
@allowed([
    '12.04.5-LTS'
    '14.04.5-LTS'
    '16.04.0-LTS'
    '18.04-LTS'
])
param ubuntuOSVersion string = '18.04-LTS'

@description('Location for all resources.')
param location string = resourceGroup().location

@description('The size of the VM')
param vmSize string = 'Standard_B2s'

@description('Name of the VNET')
param virtualNetworkName string = 'vNet'
```

```

@description('Name of the subnet in the virtual network')
param subnetName string = 'Subnet'

@description('Name of the Network Security Group')
param networkSecurityGroupName string = 'SecGroupNet'

var publicIPAddressName = '${vmName}PublicIP'
var networkInterfaceName = '${vmName}NetInt'
var osDiskType = 'Standard_LRS'
var subnetAddressPrefix = '10.1.0.0/24'
var addressPrefix = '10.1.0.0/16'
var linuxConfiguration = {
    disablePasswordAuthentication: true
    ssh: [
        {
            publicKeys: [
                {
                    path: '/home/${adminUsername}/.ssh/authorized_keys'
                    keyData: adminPasswordOrKey
                }
            ]
        }
    ]
}

resource nic 'Microsoft.Network/networkInterfaces@2021-05-01' = {
    name: networkInterfaceName
    location: location
    properties: {
        ipConfigurations: [
            {
                name: 'ipconfig1'
                properties: {
                    subnet: {
                        id: subnet.id
                    }
                    privateIPAllocationMethod: 'Dynamic'
                    publicIPAddress: {
                        id: publicIP.id
                    }
                }
            }
        ]
        networkSecurityGroup: {
            id: nsg.id
        }
    }
}

resource nsg 'Microsoft.Network/networkSecurityGroups@2021-05-01' = {
    name: networkSecurityGroupName
    location: location
    properties: {
        securityRules: [
            {
                name: 'SSH'
                properties: {
                    priority: 1000
                    protocol: 'Tcp'
                    access: 'Allow'
                    direction: 'Inbound'
                    sourceAddressPrefix: '*'
                    sourcePortRange: '*'
                    destinationAddressPrefix: '*'
                    destinationPortRange: '22'
                }
            }
        ]
    }
}

```

```

resource vnet 'Microsoft.Network/virtualNetworks@2021-05-01' = {
  name: virtualNetworkName
  location: location
  properties: {
    addressSpace: {
      addressPrefixes: [
        addressPrefix
      ]
    }
  }
}

resource subnet 'Microsoft.Network/virtualNetworks/subnets@2021-05-01' = {
  parent: vnet
  name: subnetName
  properties: {
    addressPrefix: subnetAddressPrefix
    privateEndpointNetworkPolicies: 'Enabled'
    privateLinkServiceNetworkPolicies: 'Enabled'
  }
}

resource publicIP 'Microsoft.Network/publicIPAddresses@2021-05-01' = {
  name: publicIPAddressName
  location: location
  sku: {
    name: 'Basic'
  }
  properties: {
    publicIPAllocationMethod: 'Dynamic'
    publicIPAddressVersion: 'IPv4'
    dnsSettings: {
      domainNameLabel: dnsLabelPrefix
    }
    idleTimeoutInMinutes: 4
  }
}

resource vm 'Microsoft.Compute/virtualMachines@2021-11-01' = {
  name: vmName
  location: location
  properties: {
    hardwareProfile: {
      vmSize: vmSize
    }
    storageProfile: {
      osDisk: {
        createOption: 'FromImage'
        managedDisk: {
          storageAccountType: osDiskType
        }
      }
      imageReference: {
        publisher: 'Canonical'
        offer: 'UbuntuServer'
        sku: ubuntuOSVersion
        version: 'latest'
      }
    }
    networkProfile: {
      networkInterfaces: [
        {
          id: nic.id
        }
      ]
    }
    osProfile: {
      computerName: vmName
    }
  }
}

```

```

        adminUsername: adminUsername
        adminPassword: adminPasswordOrKey
        linuxConfiguration: ((authenticationType == 'password') ? null : linuxConfiguration)
    }
}

output adminUsername string = adminUsername
output hostname string = publicIP.properties.dnsSettings.fqdn
output sshCommand string = 'ssh ${adminUsername}@${publicIP.properties.dnsSettings.fqdn}'

```

Several resources are defined in the Bicep file:

- [Microsoft.Network/virtualNetworks/subnets](#): create a subnet.
- [Microsoft.Storage/storageAccounts](#): create a storage account.
- [Microsoft.Network/networkInterfaces](#): create a NIC.
- [Microsoft.Network/networkSecurityGroups](#): create a network security group.
- [Microsoft.Network/virtualNetworks](#): create a virtual network.
- [Microsoft.Network/publicIPAddresses](#): create a public IP address.
- [Microsoft.Compute/virtualMachines](#): create a virtual machine.

## Deploy the Bicep file

1. Save the Bicep file as `main.bicep` to your local computer.
2. Deploy the Bicep file using either Azure CLI or Azure PowerShell.

- [CLI](#)
- [PowerShell](#)

```

az group create --name exampleRG --location eastus

az deployment group create --resource-group exampleRG --template-file main.bicep --parameters
adminUsername=<admin-username>

```

### NOTE

Replace `<admin-username>` with a unique username. You'll also be prompted to enter `adminPasswordOrKey`.

When the deployment finishes, you should see a message indicating the deployment succeeded.

## Review deployed resources

Use the Azure portal, Azure CLI, or Azure PowerShell to list the deployed resources in the resource group.

- [CLI](#)
- [PowerShell](#)

```

az resource list --resource-group exampleRG

```

## Clean up resources

When no longer needed, use the Azure portal, Azure CLI, or Azure PowerShell to delete the VM and all of the

resources in the resource group.

- [CLI](#)
- [PowerShell](#)

```
az group delete --name exampleRG
```

## Next steps

In this quickstart, you deployed a simple virtual machine using a Bicep file. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Create an Ubuntu Linux virtual machine using an ARM template

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This quickstart shows you how to use an Azure Resource Manager template (ARM template) to deploy an Ubuntu Linux virtual machine (VM) in Azure.

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

If your environment meets the prerequisites and you're familiar with using ARM templates, select the **Deploy to Azure** button. The template will open in the Azure portal.

 Deploy to Azure

## Prerequisites

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Review the template

The template used in this quickstart is from [Azure Quickstart Templates](#).

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "metadata": {  
    "_generator": {  
      "name": "bicep",  
      "version": "0.8.9.13224",  
      "templateHash": "15559643551456468043"  
    }  
  },  
  "parameters": {  
    "vmName": {  
      "type": "string",  
      "defaultValue": "simpleLinuxVM",  
      "metadata": {  
        "description": "The name of your Virtual Machine."  
      }  
    },  
    "adminUsername": {  
      "type": "string",  
      "metadata": {  
        "description": "Username for the Virtual Machine."  
      }  
    },  
    "authenticationType": {  
      "type": "string",  
      "defaultValue": "password",  
      "allowedValues": [  
        "sshPublicKey",  
        "password"  
      ]  
    }  
  }  
}
```

```
        ],
        "metadata": {
            "description": "Type of authentication to use on the Virtual Machine. SSH key is recommended."
        }
    },
    "adminPasswordOrKey": {
        "type": "secureString",
        "metadata": {
            "description": "SSH Key or password for the Virtual Machine. SSH key is recommended."
        }
    },
    "dnsLabelPrefix": {
        "type": "string",
        "defaultValue": "[toLowerCase(format('{0}-{1}', parameters('vmName'),
uniqueString(resourceGroup().id)))]",
        "metadata": {
            "description": "Unique DNS Name for the Public IP used to access the Virtual Machine."
        }
    },
    "ubuntuOSVersion": {
        "type": "string",
        "defaultValue": "18.04-LTS",
        "allowedValues": [
            "12.04.5-LTS",
            "14.04.5-LTS",
            "16.04.0-LTS",
            "18.04-LTS"
        ],
        "metadata": {
            "description": "The Ubuntu version for the VM. This will pick a fully patched image of this given
Ubuntu version."
        }
    },
    "location": {
        "type": "string",
        "defaultValue": "[resourceGroup().location]",
        "metadata": {
            "description": "Location for all resources."
        }
    },
    "vmSize": {
        "type": "string",
        "defaultValue": "Standard_B2s",
        "metadata": {
            "description": "The size of the VM"
        }
    },
    "virtualNetworkName": {
        "type": "string",
        "defaultValue": "vNet",
        "metadata": {
            "description": "Name of the VNET"
        }
    },
    "subnetName": {
        "type": "string",
        "defaultValue": "Subnet",
        "metadata": {
            "description": "Name of the subnet in the virtual network"
        }
    },
    "networkSecurityGroupName": {
        "type": "string",
        "defaultValue": "SecGroupNet",
        "metadata": {
            "description": "Name of the Network Security Group"
        }
    }
},
```

```

"variables": {
    "publicIPAttributeName": "[format('{0}PublicIP', parameters('vmName'))]",
    "networkInterfaceName": "[format('{0}NetInt', parameters('vmName'))]",
    "osDiskType": "Standard_LRS",
    "subnetAddressPrefix": "10.1.0.0/24",
    "addressPrefix": "10.1.0.0/16",
    "linuxConfiguration": {
        "disablePasswordAuthentication": true,
        "ssh": {
            "publicKeys": [
                {
                    "path": "[format('/home/{0}/.ssh/authorized_keys', parameters('adminUsername'))]",
                    "keyData": "[parameters('adminPasswordOrKey')]"
                }
            ]
        }
    }
},
"resources": [
{
    "type": "Microsoft.Network/networkInterfaces",
    "apiVersion": "2021-05-01",
    "name": "[variables('networkInterfaceName')]",
    "location": "[parameters('location')]",
    "properties": {
        "ipConfigurations": [
            {
                "name": "ipconfig1",
                "properties": {
                    "subnet": {
                        "id": "[resourceId('Microsoft.Network/virtualNetworks/subnets',
parameters('virtualNetworkName'), parameters('subnetName'))]"
                    },
                    "privateIPAllocationMethod": "Dynamic",
                    "publicIPAddress": {
                        "id": "[resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))]"
                    }
                }
            }
        ],
        "networkSecurityGroup": {
            "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
parameters('networkSecurityGroupName'))]"
        }
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkSecurityGroups', parameters('networkSecurityGroupName'))]",
        "[resourceId('Microsoft.Network/publicIPAddresses', variables('publicIPAddressName'))]",
        "[resourceId('Microsoft.Network/virtualNetworks/subnets', parameters('virtualNetworkName'),
parameters('subnetName'))]"
    ]
},
{
    "type": "Microsoft.Network/networkSecurityGroups",
    "apiVersion": "2021-05-01",
    "name": "[parameters('networkSecurityGroupName')]",
    "location": "[parameters('location')]",
    "properties": {
        "securityRules": [
            {
                "name": "SSH",
                "properties": {
                    "priority": 1000,
                    "protocol": "Tcp",
                    "access": "Allow",
                    "direction": "Inbound",
                    "sourceAddressPrefix": "*",
                    "sourcePortRange": "*",
                    "destinationAddressPrefix": "*",
                    "destinationPortRange": "*"
                }
            }
        ]
    }
}
]

```

```

        "destinationAddressPrefix": "*",
        "destinationPortRange": "22"
    }
}
]
}
},
{
    "type": "Microsoft.Network/virtualNetworks",
    "apiVersion": "2021-05-01",
    "name": "[parameters('virtualNetworkName')]",
    "location": "[parameters('location')]",
    "properties": {
        "addressSpace": {
            "addressPrefixes": [
                "[variables('addressPrefix')]"
            ]
        }
    }
},
{
    "type": "Microsoft.Network/virtualNetworks/subnets",
    "apiVersion": "2021-05-01",
    "name": "[format('{0}/{1}', parameters('virtualNetworkName'), parameters('subnetName'))]",
    "properties": {
        "addressPrefix": "[variables('subnetAddressPrefix')]",
        "privateEndpointNetworkPolicies": "Enabled",
        "privateLinkServiceNetworkPolicies": "Enabled"
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/virtualNetworks', parameters('virtualNetworkName'))]"
    ]
},
{
    "type": "Microsoft.Network/publicIPAddresses",
    "apiVersion": "2021-05-01",
    "name": "[variables('publicIPAddressName')]",
    "location": "[parameters('location')]",
    "sku": {
        "name": "Basic"
    },
    "properties": {
        "publicIPAllocationMethod": "Dynamic",
        "publicIPAddressVersion": "IPv4",
        "dnsSettings": {
            "domainNameLabel": "[parameters('dnsLabelPrefix')]"
        },
        "idleTimeoutInMinutes": 4
    }
},
{
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2021-11-01",
    "name": "[parameters('vmName')]",
    "location": "[parameters('location')]",
    "properties": {
        "hardwareProfile": {
            "vmSize": "[parameters('vmSize')]"
        },
        "storageProfile": {
            "osDisk": {
                "createOption": "FromImage",
                "managedDisk": {
                    "storageAccountType": "[variables('osDiskType')]"
                }
            },
            "imageReference": {
                "publisher": "Canonical",
                "offer": "UbuntuServer",
                "version": "latest"
            }
        }
    }
}
]
```

```

        "sku": "[parameters('ubuntuOSVersion')]",
        "version": "latest"
    },
    "networkProfile": {
        "networkInterfaces": [
            {
                "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]"
            }
        ],
        "osProfile": {
            "computerName": "[parameters('vmName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "adminPassword": "[parameters('adminPasswordOrKey')]",
            "linuxConfiguration": "[if>equals(parameters('authenticationType'), 'password'), null(),
variables('linuxConfiguration'))]"
        }
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]"
    ]
},
"outputs": {
    "adminUsername": {
        "type": "string",
        "value": "[parameters('adminUsername')]"
    },
    "hostname": {
        "type": "string",
        "value": "[reference(resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))).dnsSettings.fqdn]"
    },
    "sshCommand": {
        "type": "string",
        "value": "[format('ssh {0}@{1}', parameters('adminUsername'),
reference(resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))).dnsSettings.fqdn)]"
    }
}
}

```

Several resources are defined in the template:

- **Microsoft.Network/virtualNetworks/subnets**: create a subnet.
- **Microsoft.Storage/storageAccounts**: create a storage account.
- **Microsoft.Network/networkInterfaces**: create a NIC.
- **Microsoft.Network/networkSecurityGroups**: create a network security group.
- **Microsoft.Network/virtualNetworks**: create a virtual network.
- **Microsoft.Network/publicIPAddresses**: create a public IP address.
- **Microsoft.Compute/virtualMachines**: create a virtual machine.

## Deploy the template

1. Select the following image to sign in to Azure and open a template. The template creates a key vault and a secret.



2. Select or enter the following values. Use the default values, when available.

- **Subscription:** select an Azure subscription.
- **Resource group:** select an existing resource group from the drop-down, or select **Create new**, enter a unique name for the resource group, and then click **OK**.
- **Location:** select a location. For example, **Central US**.
- **Admin username:** provide a username, such as *azureuser*.
- **Authentication type:** You can choose between using an SSH key or a password.
- **Admin Password Or Key** depending on what you choose for authentication type:
  - If you choose **password**, the password must be at least 12 characters long and meet the [defined complexity requirements](#).
  - If you choose **sshPublicKey**, paste in the contents of your public key.
- **DNS label prefix:** enter a unique identifier to use as part of the DNS label.
- **Ubuntu OS version:** select which version of Ubuntu you want to run on the VM.
- **Location:** the default is the same location as the resource group, if it already exists.
- **VM size:** select the [size](#) to use for the VM.
- **Virtual Network Name:** name to be used for the vNet.
- **Subnet Name:** name for the subnet the VM should use.
- **Network Security Group Name:** name for the NSG.

3. Select **Review + create**. After validation completes, select **Create** to create and deploy the VM.

The Azure portal is used to deploy the template. In addition to the Azure portal, you can also use the Azure CLI, Azure PowerShell, and REST API. To learn other deployment methods, see [Deploy templates](#).

## Review deployed resources

You can use the Azure portal to check on the VM and other resource that were created. After the deployment is finished, select **Go to resource group** to see the VM and other resources.

## Clean up resources

When no longer needed, delete the resource group, which deletes the VM and all of the resources in the resource group.

1. Select the **Resource group**.
2. On the page for the resource group, select **Delete**.
3. When prompted, type the name of the resource group and then select **Delete**.

## Next steps

In this quickstart, you deployed a simple virtual machine using an ARM template. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Linux virtual machine tutorials](#)

# Quickstart: Create a Windows virtual machine with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The Azure CLI is used to create and manage Azure resources from the command line or in scripts. This quickstart shows you how to use the Azure CLI to deploy a virtual machine (VM) in Azure that runs Windows Server 2019. To see your VM in action, you then RDP to the VM and install the IIS web server.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/bash>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

## Create a resource group

Create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name myResourceGroup --location eastus
```

## Create virtual machine

Create a VM with [az vm create](#). The following example creates a VM named *myVM*. This example uses *azureuser* for an administrative user name.

You will need to supply a password that meets the [password requirements for Azure VMs](#).

Using the example below, you will be prompted to enter a password at the command line. You could also add the the `--admin-password` parameter with a value for your password. The user name and password will be used later, when you connect to the VM.

```
az vm create \
  --resource-group myResourceGroup \
  --name myVM \
  --image Win2022AzureEditionCore \
  --public-ip-sku Standard \
  --admin-username azureuser
```

It takes a few minutes to create the VM and supporting resources. The following example output shows the VM create operation was successful.

```
{  
  "fqdns": "",  
  "id":  
    "/subscriptions/<guid>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",  
  "location": "eastus",  
  "macAddress": "00-0D-3A-23-9A-49",  
  "powerState": "VM running",  
  "privateIpAddress": "10.0.0.4",  
  "publicIpAddress": "52.174.34.95",  
  "resourceGroup": "myResourceGroup"  
}
```

Note your own `publicIpAddress` in the output from your VM. This address is used to access the VM in the next steps.

## Install web server

To see your VM in action, install the IIS web server.

```
az vm run-command invoke -g MyResourceGroup -n MyVm --command-id RunPowerShellScript --scripts "Install-WindowsFeature -name Web-Server -IncludeManagementTools"
```

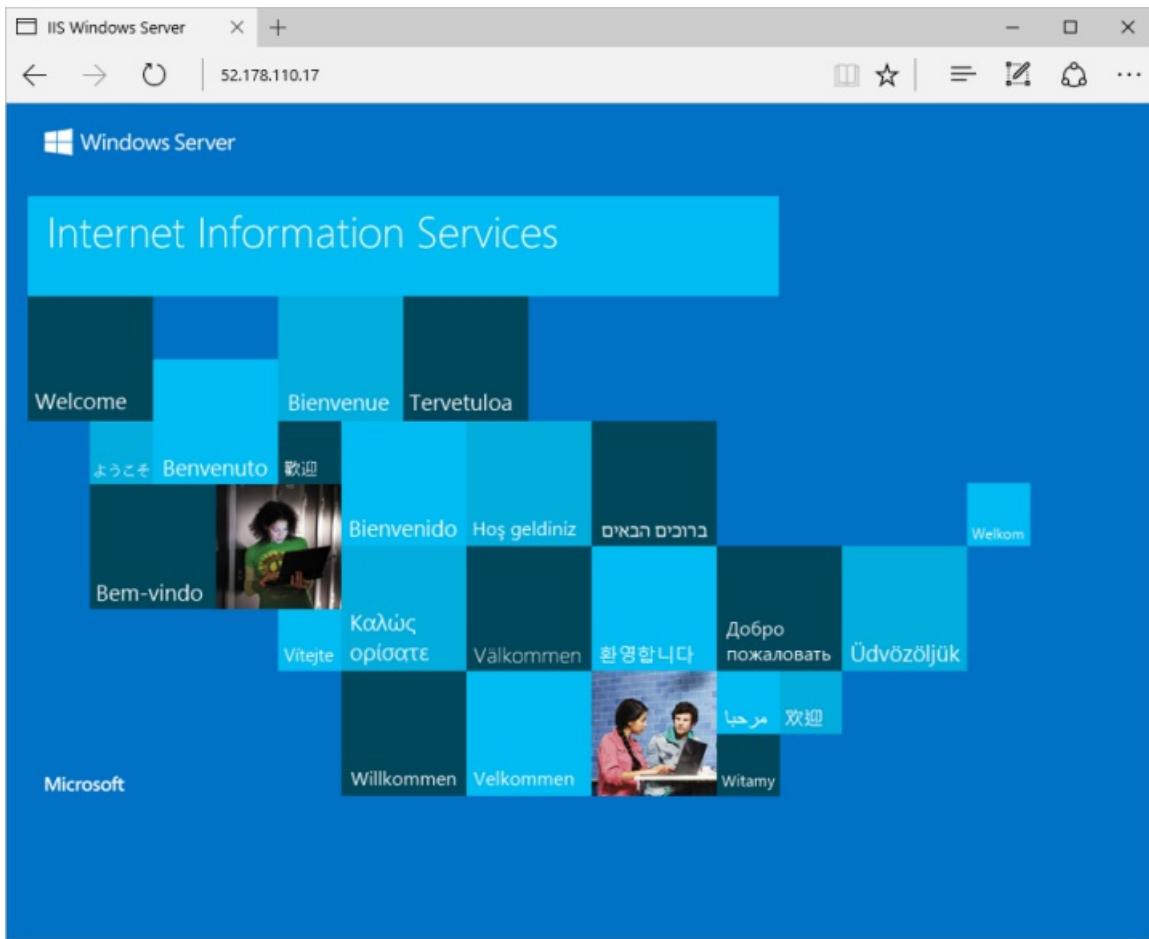
## Open port 80 for web traffic

By default, only RDP connections are opened when you create a Windows VM in Azure. Use `az vm open-port` to open TCP port 80 for use with the IIS web server:

```
az vm open-port --port 80 --resource-group myResourceGroup --name myVM
```

## View the web server in action

With IIS installed and port 80 now open on your VM from the Internet, use a web browser of your choice to view the default IIS welcome page. Use the public IP address of your VM obtained in a previous step. The following example shows the default IIS web site:



## Clean up resources

When no longer needed, you can use the [az group delete](#) command to remove the resource group, VM, and all related resources:

```
az group delete --name myResourceGroup
```

## Next steps

In this quickstart, you deployed a simple virtual machine, open a network port for web traffic, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Windows VMs.

[Azure Windows virtual machine tutorials](#)

# Quickstart: Create a Windows virtual machine in the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Azure virtual machines (VMs) can be created through the Azure portal. This method provides a browser-based user interface to create VMs and their associated resources. This quickstart shows you how to use the Azure portal to deploy a virtual machine (VM) in Azure that runs Windows Server 2019. To see your VM in action, you then RDP to the VM and install the IIS web server.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Sign in to Azure

Sign in to the Azure portal at <https://portal.azure.com>.

## Create virtual machine

1. Enter *virtual machines* in the search.
2. Under **Services**, select **Virtual machines**.
3. In the **Virtual machines** page, select **Create** and then **Azure virtual machine**. The **Create a virtual machine** page opens.
4. Under **Instance details**, enter *myVM* for the **Virtual machine name** and choose **Windows Server 2019 Datacenter - Gen 2** for the **Image**. Leave the other defaults.

Instance details

Virtual machine name *	myVM
Region *	(US) West US
Availability options	No infrastructure redundancy required
Security type	Standard
Image *	<input checked="" type="checkbox"/> Windows Server 2019 Datacenter - Gen2 <a href="#">See all images</a>   <a href="#">Configure VM generation</a>
VM architecture	<input type="radio"/> Arm64 <input checked="" type="radio"/> x64  <small>ⓘ Arm64 is not supported with the selected image.</small>

### NOTE

Some users will now see the option to create VMs in multiple zones. To learn more about this new capability, see [Create virtual machines in an availability zone](#).

Availability zone *	Zones 1
<small>⚡ You can now select multiple zones. Selecting multiple zones will create one VM per zone.</small>	

5. Under **Administrator account**, provide a username, such as *azureuser* and a password. The password must be at least 12 characters long and meet the [defined complexity requirements](#).

Administrator account

Username * ⓘ	azureuser	✓
Password * ⓘ	*****	✓
Confirm password * ⓘ	*****	✓

6. Under **Inbound port rules**, choose **Allow selected ports** and then select **RDP (3389)** and **HTTP (80)** from the drop-down.

**Inbound port rules**

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports \* ⓘ

None  
 Allow selected ports

Select inbound ports \*

RDP (3389) ▾

**⚠ This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

7. Leave the remaining defaults and then select the **Review + create** button at the bottom of the page.

**Licensing**

Save up to 49% with a license you already own using Azure Hybrid Benefit. [Learn more ↗](#)

Would you like to use an existing Windows Server license? \* ⓘ

[Review Azure hybrid benefit compliance](#)

**Review + create**

< Previous

Next : Disks >

8. After validation runs, select the **Create** button at the bottom of the page.

## Create a virtual machine

Validation passed

### Basics

Subscription	myAzureSubscription
Resource group	(new) myVM_group_08290738
Virtual machine name	myVM
Region	East US
Availability options	No infrastructure redundancy required
Security type	Standard
Image	Windows Server 2022 Datacenter: Azure Edition - Gen2
VM architecture	x64
Size	Standard D2s v3 (2 vcpus, 8 GiB memory)
Username	azureuser
Public inbound ports	RDP

Already have a Windows license?  No

**Create**

< Previous

Next >

Download a template for automation

- After deployment is complete, select **Go to resource**.

### Next steps

- [Setup auto-shutdown](#) Recommended
- [Monitor VM health, performance and network dependencies](#) Recommended
- [Run a script inside the virtual machine](#) Recommended

**Go to resource**

[Create another VM](#)

## Connect to virtual machine

Create a remote desktop connection to the virtual machine. These directions tell you how to connect to your VM from a Windows computer. On a Mac, you need an RDP client such as this [Remote Desktop Client](#) from the Mac App Store.

- On the overview page for your virtual machine, select the **Connect > RDP**.

The screenshot shows the Azure portal interface for a virtual machine named 'myVM'. At the top, there's a navigation bar with 'Home > myVM'. Below it is a card for 'myVM' with a 'Virtual machine' icon. A search bar labeled 'Search (Ctrl+ /)' is on the left. To its right is a 'Connect' button, which is highlighted with a red box. Other buttons in the top row include 'Start', 'Restart', 'Stop', 'Capture', 'Delete', and 'Refresh'. Below the search bar is a 'Overview' section with tabs for 'Overview' and 'Activity log'. The 'Overview' tab is selected. It displays the following details: Resource group: myResourceGroup (with a 'Change' link), Status: Running, and Location: East US.

- In the **Connect with RDP** tab, keep the default options to connect by IP address, over port 3389, and click **Download RDP file**.

## Connect with RDP

✓ Suggested method for connecting

To connect to your virtual machine via RDP, select an IP address, optionally change the port number, and download the RDP file.

IP address \*

Public IP address (192.168.1.253)

Port number \*

3389

**Download RDP File**

3. Open the downloaded RDP file and click **Connect** when prompted.
4. In the **Windows Security** window, select **More choices** and then **Use a different account**. Type the username as `localhost\username`, enter the password you created for the virtual machine, and then click **OK**.
5. You may receive a certificate warning during the sign-in process. Click **Yes** or **Continue** to create the connection.

## Install web server

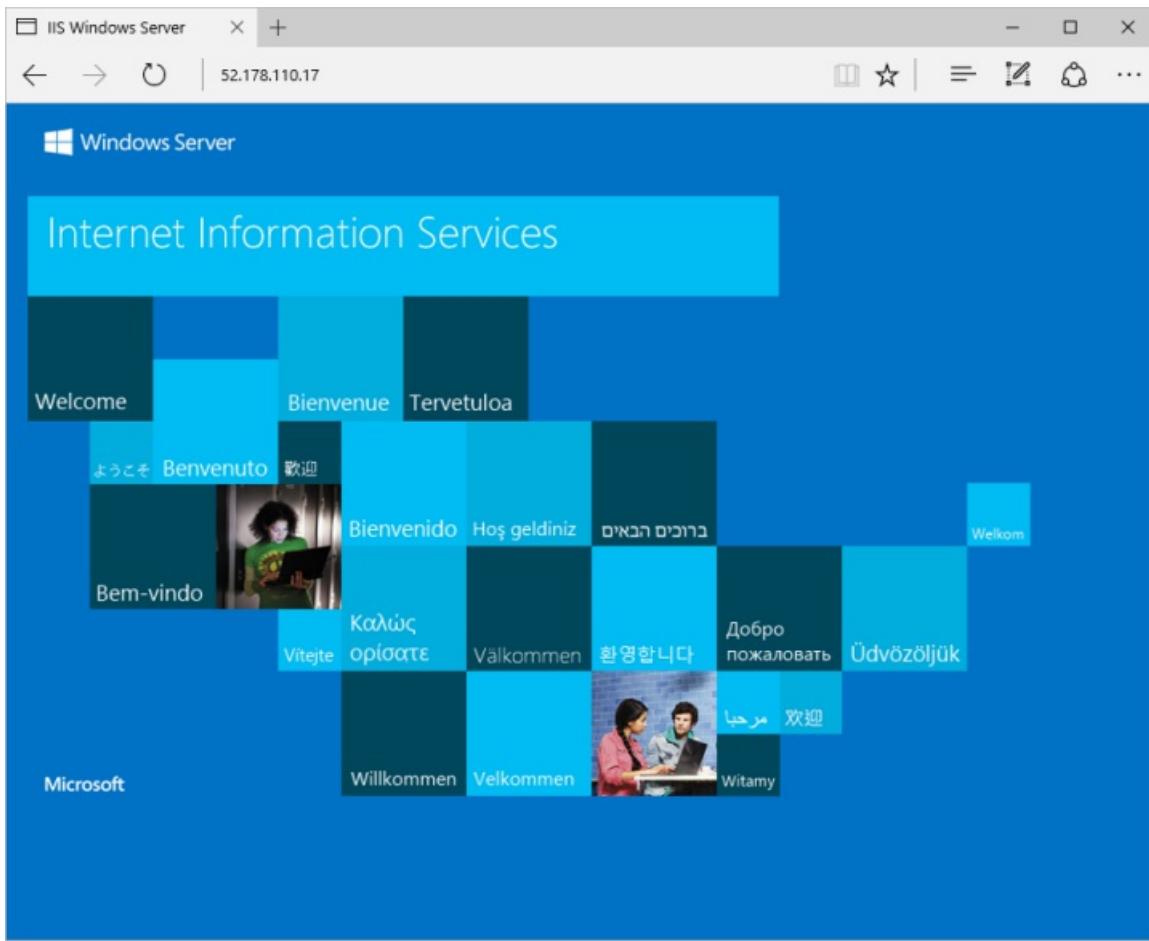
To see your VM in action, install the IIS web server. Open a PowerShell prompt on the VM and run the following command:

```
Install-WindowsFeature -name Web-Server -IncludeManagementTools
```

When done, close the RDP connection to the VM.

## View the IIS welcome page

In the portal, select the VM and in the overview of the VM, hover over the IP address to show **Copy to clipboard**. Copy the IP address and paste it into a browser tab. The default IIS welcome page will open, and should look like this:



## Clean up resources

When no longer needed, you can delete the resource group, virtual machine, and all related resources.

1. On the Overview page for the VM, select the **Resource group** link.
2. At the top of the page for the resource group, select **Delete resource group**.
3. A page will open warning you that you are about to delete resources. Type the name of the resource group and select **Delete** to finish deleting the resources and the resource group.

## Next steps

In this quickstart, you deployed a simple virtual machine, opened a network port for web traffic, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Windows VMs.

[Azure Windows virtual machine tutorials](#)

# Quickstart: Create a Windows virtual machine in Azure with PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The Azure PowerShell module is used to create and manage Azure resources from the PowerShell command line or in scripts. This quickstart shows you how to use the Azure PowerShell module to deploy a virtual machine (VM) in Azure that runs Windows Server 2016. You will also RDP to the VM and install the IIS web server, to show the VM in action.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create resource group

Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which Azure resources are deployed and managed.

```
New-AzResourceGroup -Name 'myResourceGroup' -Location 'EastUS'
```

## Create virtual machine

Create a VM with [New-AzVm](#). Provide names for each of the resources and the `New-AzVm` cmdlet creates if they don't already exist.

When prompted, provide a username and password to be used as the sign-in credentials for the VM:

```
New-AzVm ` 
-ResourceGroupName 'myResourceGroup' ` 
-Name 'myVM' ` 
-Location 'East US' ` 
-VirtualNetworkName 'myVnet' ` 
-SubnetName 'mySubnet' ` 
-SecurityGroupName 'myNetworkSecurityGroup' ` 
-PublicIpAddressName 'myPublicIpAddress' ` 
-OpenPorts 80,3389
```

## Install web server

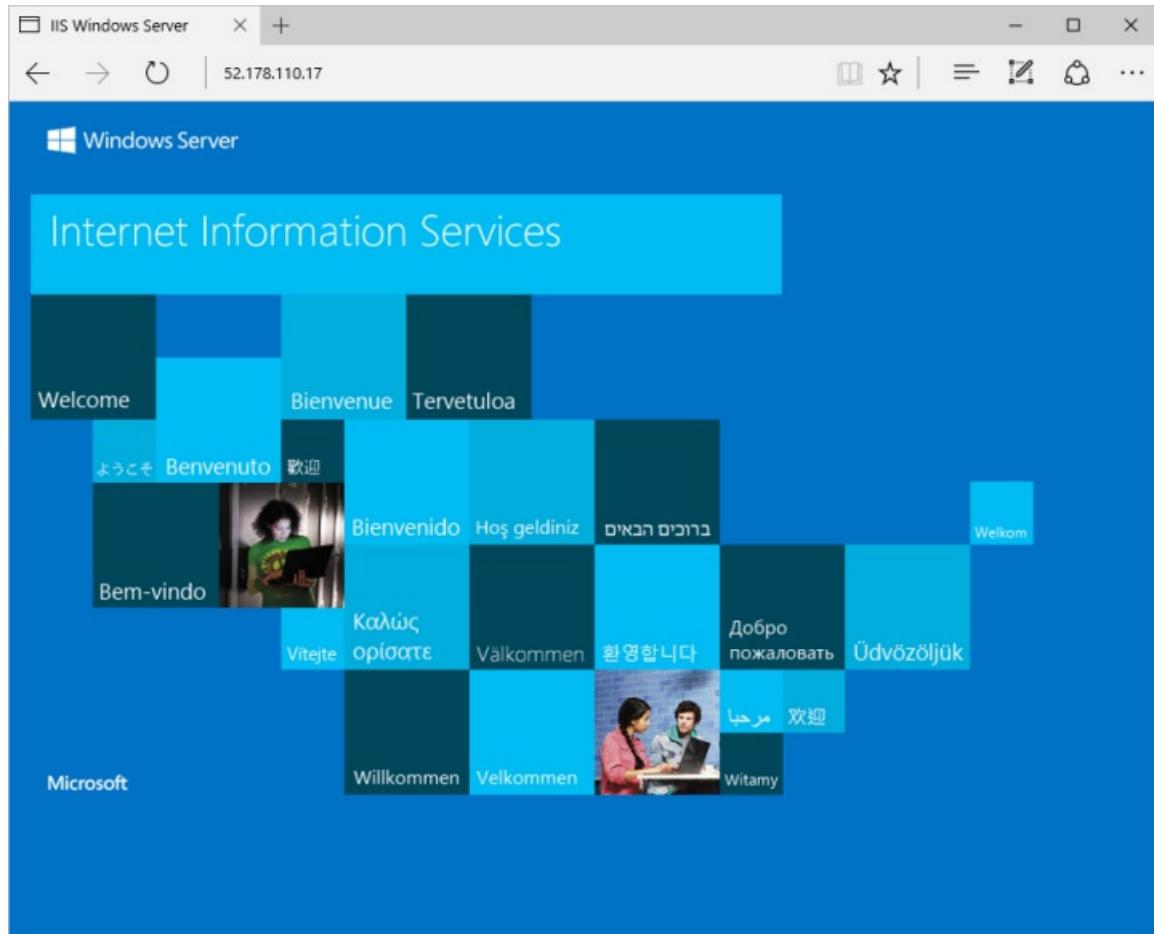
To see your VM in action, install the IIS web server. Open a PowerShell prompt on the VM and run the following command:

```
Invoke-AzVMRunCommand -ResourceGroupName 'myResourceGroup' -VMName 'myVM' -CommandId 'RunPowerShellScript' -  
ScriptString  
'Install-WindowsFeature -Name Web-Server -IncludeManagementTools'
```

The `-ScriptString` parameter requires version `4.27.0` or later of the `Az.Compute` module.

## View the web server in action

With IIS installed and port 80 now open on your VM from the Internet, use a web browser of your choice to view the default IIS welcome page. Use the public IP address of your VM obtained in a previous step. The following example shows the default IIS web site:



## Clean up resources

When no longer needed, you can use the [Remove-AzResourceGroup](#) cmdlet to remove the resource group, VM, and all related resources:

```
Remove-AzResourceGroup -Name 'myResourceGroup'
```

## Next steps

In this quickstart, you deployed a simple virtual machine, opened a network port for web traffic, and installed a basic web server. To learn more about Azure virtual machines, continue to the tutorial for Windows VMs.

[Azure Windows virtual machine tutorials](#)

# Quickstart: Create a Windows virtual machine using a Bicep file

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This quickstart shows you how to use a Bicep file to deploy a Windows virtual machine (VM) in Azure.

[Bicep](#) is a domain-specific language (DSL) that uses declarative syntax to deploy Azure resources. It provides concise syntax, reliable type safety, and support for code reuse. Bicep offers the best authoring experience for your infrastructure-as-code solutions in Azure.

## Prerequisites

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Review the Bicep file

The Bicep file used in this quickstart is from [Azure Quickstart Templates](#).

```
@description('Username for the Virtual Machine.')
param adminUsername string

@description('Password for the Virtual Machine.')
@param minLength(12)
@secure()
param adminPassword string

@description('Unique DNS Name for the Public IP used to access the Virtual Machine.')
param dnsLabelPrefix string = toLower('${vmName}-${uniqueString(resourceGroup().id, vmName)}')

@description('Name for the Public IP used to access the Virtual Machine.')
param publicIpName string = 'myPublicIP'

@description('Allocation method for the Public IP used to access the Virtual Machine.')
@allowed([
    'Dynamic'
    'Static'
])
param publicIPAllocationMethod string = 'Dynamic'

@description('SKU for the Public IP used to access the Virtual Machine.')
@allowed([
    'Basic'
    'Standard'
])
param publicIpSku string = 'Basic'

@description('The Windows version for the VM. This will pick a fully patched image of this given Windows
version.')
@allowed([
    '2008-R2-SP1'
    '2008-R2-SP1-smalldisk'
    '2012-Datacenter'
    '2012-datacenter-gensecond'
    '2012-Datacenter-smalldisk'
    '2012-datacenter-smalldisk-g2'
    '2012-Datacenter-zhcn'
```

```

'2012-datacenter-zhcn-g2'
'2012-R2-Datacenter'
'2012-r2-datacenter-gensecond'
'2012-R2-Datacenter-smalldisk'
'2012-r2-datacenter-smalldisk-g2'
'2012-R2-Datacenter-zhcn'
'2012-r2-datacenter-zhcn-g2'
'2016-Datacenter'
'2016-datacenter-gensecond'
'2016-datacenter-gs'
'2016-Datacenter-Server-Core'
'2016-datacenter-server-core-g2'
'2016-Datacenter-Server-Core-smalldisk'
'2016-datacenter-server-core-smalldisk-g2'
'2016-Datacenter-smalldisk'
'2016-datacenter-smalldisk-g2'
'2016-Datacenter-with-Containers'
'2016-datacenter-with-containers-g2'
'2016-datacenter-with-containers-gs'
'2016-Datacenter-zhcn'
'2016-datacenter-zhcn-g2'
'2019-Datacenter'
'2019-Datacenter-Core'
'2019-datacenter-core-g2'
'2019-Datacenter-Core-smalldisk'
'2019-datacenter-core-smalldisk-g2'
'2019-Datacenter-Core-with-Containers'
'2019-datacenter-core-with-containers-g2'
'2019-Datacenter-Core-with-Containers-smalldisk'
'2019-datacenter-core-with-containers-smalldisk-g2'
'2019-datacenter-gensecond'
'2019-datacenter-gs'
'2019-Datacenter-smalldisk'
'2019-datacenter-smalldisk-g2'
'2019-Datacenter-with-Containers'
'2019-datacenter-with-containers-g2'
'2019-datacenter-with-containers-gs'
'2019-Datacenter-with-Containers-smalldisk'
'2019-datacenter-with-containers-smalldisk-g2'
'2019-Datacenter-zhcn'
'2019-datacenter-zhcn-g2'
'2022-datacenter'
'2022-datacenter-azure-edition'
'2022-datacenter-azure-edition-core'
'2022-datacenter-azure-edition-core-smalldisk'
'2022-datacenter-azure-edition-smalldisk'
'2022-datacenter-core'
'2022-datacenter-core-g2'
'2022-datacenter-core-smalldisk'
'2022-datacenter-core-smalldisk-g2'
'2022-datacenter-g2'
'2022-datacenter-smalldisk'
'2022-datacenter-smalldisk-g2'
])
param OSVersion string = '2022-datacenter-azure-edition-core'

@description('Size of the virtual machine.')
param vmSize string = 'Standard_D2s_v5'

@description('Location for all resources.')
param location string = resourceGroup().location

@description('Name of the virtual machine.')
param vmName string = 'simple-vm'

var storageAccountName = 'bootdiags${uniqueString(resourceGroup().id)}'
var nicName = 'myVMNic'
var addressPrefix = '10.0.0.0/16'
var subnetName = 'Subnet'

```

```

var subnetPrefix = '10.0.0.0/24'
var virtualNetworkName = 'MyVNET'
var networkSecurityGroupName = 'default-NSG'

resource stg 'Microsoft.Storage/storageAccounts@2021-04-01' = {
  name: storageAccountName
  location: location
  sku: {
    name: 'Standard_LRS'
  }
  kind: 'Storage'
}

resource pip 'Microsoft.Network/publicIPAddresses@2021-02-01' = {
  name: publicIpName
  location: location
  sku: {
    name: publicIpSku
  }
  properties: {
    publicIPAllocationMethod: publicIPAllocationMethod
    dnsSettings: {
      domainNameLabel: dnsLabelPrefix
    }
  }
}

resource securityGroup 'Microsoft.Network/networkSecurityGroups@2021-02-01' = {
  name: networkSecurityGroupName
  location: location
  properties: {
    securityRules: [
      {
        name: 'default-allow-3389'
        properties: {
          priority: 1000
          access: 'Allow'
          direction: 'Inbound'
          destinationPortRange: '3389'
          protocol: 'Tcp'
          sourcePortRange: '*'
          sourceAddressPrefix: '*'
          destinationAddressPrefix: '*'
        }
      }
    ]
  }
}

resource vn 'Microsoft.Network/virtualNetworks@2021-02-01' = {
  name: virtualNetworkName
  location: location
  properties: {
    addressSpace: {
      addressPrefixes: [
        addressPrefix
      ]
    }
    subnets: [
      {
        name: subnetName
        properties: {
          addressPrefix: subnetPrefix
          networkSecurityGroup: {
            id: securityGroup.id
          }
        }
      }
    ]
  }
}

```

```

    }

resource nic 'Microsoft.Network/networkInterfaces@2021-02-01' = {
  name: nicName
  location: location
  properties: {
    ipConfigurations: [
      {
        name: 'ipconfig1'
        properties: {
          privateIPAllocationMethod: 'Dynamic'
          publicIPAddress: {
            id: pip.id
          }
          subnet: {
            id: resourceId('Microsoft.Network/virtualNetworks/subnets', vn.name, subnetName)
          }
        }
      }
    ]
  }
}

resource vm 'Microsoft.Compute/virtualMachines@2021-03-01' = {
  name: vmName
  location: location
  properties: {
    hardwareProfile: {
      vmSize: vmSize
    }
    osProfile: {
      computerName: vmName
      adminUsername: adminUsername
      adminPassword: adminPassword
    }
    storageProfile: {
      imageReference: {
        publisher: 'MicrosoftWindowsServer'
        offer: 'WindowsServer'
        sku: OSVersion
        version: 'latest'
      }
      osDisk: {
        createOption: 'FromImage'
        managedDisk: {
          storageAccountType: 'StandardSSD_LRS'
        }
      }
      dataDisks: [
        {
          diskSizeGB: 1023
          lun: 0
          createOption: 'Empty'
        }
      ]
    }
    networkProfile: {
      networkInterfaces: [
        {
          id: nic.id
        }
      ]
    }
    diagnosticsProfile: {
      bootDiagnostics: {
        enabled: true
        storageUri: stg.properties.primaryEndpoints.blob
      }
    }
  }
}

```

```
        }
    }
}

output hostname string = pip.properties.dnsSettings.fqdn
```

Several resources are defined in the Bicep file:

- [Microsoft.Network/virtualNetworks/subnets](#): create a subnet.
- [Microsoft.Storage/storageAccounts](#): create a storage account.
- [Microsoft.Network/publicIPAddresses](#): create a public IP address.
- [Microsoft.Network/networkSecurityGroups](#): create a network security group.
- [Microsoft.Network/virtualNetworks](#): create a virtual network.
- [Microsoft.Network/networkInterfaces](#): create a NIC.
- [Microsoft.Compute/virtualMachines](#): create a virtual machine.

## Deploy the Bicep file

1. Save the Bicep file as **main.bicep** to your local computer.
2. Deploy the Bicep file using either Azure CLI or Azure PowerShell.
  - [CLI](#)
  - [PowerShell](#)

```
az group create --name exampleRG --location eastus
az deployment group create --resource-group exampleRG --template-file main.bicep --parameters
adminUsername=<admin-username>
```

### NOTE

Replace <**admin-username**> with a unique username. You'll also be prompted to enter adminPassword. The minimum password length is 12 characters.

When the deployment finishes, you should see a message indicating the deployment succeeded.

## Review deployed resources

Use the Azure portal, Azure CLI, or Azure PowerShell to list the deployed resources in the resource group.

- [CLI](#)
- [PowerShell](#)

```
az resource list --resource-group exampleRG
```

## Clean up resources

When no longer needed, use the Azure portal, Azure CLI, or Azure PowerShell to delete the VM and all of the resources in the resource group.

- [CLI](#)

- [PowerShell](#)

```
az group delete --name exampleRG
```

## Next steps

In this quickstart, you deployed a simple virtual machine using a Bicep file. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Windows virtual machine tutorials](#)

# Quickstart: Create a Windows virtual machine using an ARM template

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This quickstart shows you how to use an Azure Resource Manager template (ARM template) to deploy a Windows virtual machine (VM) in Azure.

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

If your environment meets the prerequisites and you're familiar with using ARM templates, select the **Deploy to Azure** button. The template will open in the Azure portal.

 Deploy to Azure

## Prerequisites

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Review the template

The template used in this quickstart is from [Azure Quickstart Templates](#).

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "metadata": {  
    "_generator": {  
      "name": "bicep",  
      "version": "0.8.9.13224",  
      "templateHash": "15495738823141086515"  
    }  
  },  
  "parameters": {  
    "adminUsername": {  
      "type": "string",  
      "metadata": {  
        "description": "Username for the Virtual Machine."  
      }  
    },  
    "adminPassword": {  
      "type": "secureString",  
      "minLength": 12,  
      "metadata": {  
        "description": "Password for the Virtual Machine."  
      }  
    },  
    "dnsLabelPrefix": {  
      "type": "string",  
      "defaultValue": "[toLower(format('{0}-{1}', parameters('vmName'), uniqueString(resourceGroup().id, parameters('vmName'))))]",  
      "metadata": {  
        "description": "Unique DNS Name for the Public IP used to access the Virtual Machine."  
      }  
    }  
  }  
}
```

```
        },
        "publicIpName": {
            "type": "string",
            "defaultValue": "myPublicIP",
            "metadata": {
                "description": "Name for the Public IP used to access the Virtual Machine."
            }
        },
        "publicIPAllocationMethod": {
            "type": "string",
            "defaultValue": "Dynamic",
            "allowedValues": [
                "Dynamic",
                "Static"
            ],
            "metadata": {
                "description": "Allocation method for the Public IP used to access the Virtual Machine."
            }
        },
        "publicIpSku": {
            "type": "string",
            "defaultValue": "Basic",
            "allowedValues": [
                "Basic",
                "Standard"
            ],
            "metadata": {
                "description": "SKU for the Public IP used to access the Virtual Machine."
            }
        },
        "OSVersion": {
            "type": "string",
            "defaultValue": "2022-datacenter-azure-edition-core",
            "allowedValues": [
                "2008-R2-SP1",
                "2008-R2-SP1-smalldisk",
                "2012-Datacenter",
                "2012-datacenter-gensecond",
                "2012-Datacenter-smalldisk",
                "2012-datacenter-smalldisk-g2",
                "2012-Datacenter-zhcn",
                "2012-datacenter-zhcn-g2",
                "2012-R2-Datacenter",
                "2012-r2-datacenter-gensecond",
                "2012-R2-Datacenter-smalldisk",
                "2012-r2-datacenter-smalldisk-g2",
                "2012-R2-Datacenter-zhcn",
                "2012-r2-datacenter-zhcn-g2",
                "2016-Datacenter",
                "2016-datacenter-gensecond",
                "2016-datacenter-gs",
                "2016-Datacenter-Core",
                "2016-datacenter-server-core-g2",
                "2016-Datacenter-Server-Core-smalldisk",
                "2016-datacenter-server-core-smalldisk-g2",
                "2016-Datacenter-smalldisk",
                "2016-datacenter-smalldisk-g2",
                "2016-Datacenter-with-Containers",
                "2016-datacenter-with-containers-g2",
                "2016-datacenter-with-containers-gs",
                "2016-Datacenter-zhcn",
                "2016-datacenter-zhcn-g2",
                "2019-Datacenter",
                "2019-Datacenter-Core",
                "2019-datacenter-core-g2",
                "2019-Datacenter-Core-smalldisk",
                "2019-datacenter-core-smalldisk-g2",
                "2019-Datacenter-Core-with-Containers",
                "2019-Datacenter-Core-with-Containers-g2"
            ]
        }
    }
}
```

```

    "2019-datacenter-core-with-containers-g2",
    "2019-Datacenter-Core-with-Containers-smalldisk",
    "2019-datacenter-core-with-containers-smalldisk-g2",
    "2019-datacenter-gensecond",
    "2019-datacenter-gs",
    "2019-Datacenter-smalldisk",
    "2019-datacenter-smalldisk-g2",
    "2019-Datacenter-with-Containers",
    "2019-datacenter-with-containers-g2",
    "2019-datacenter-with-containers-gs",
    "2019-Datacenter-with-Containers-smalldisk",
    "2019-datacenter-with-containers-smalldisk-g2",
    "2019-Datacenter-zhcn",
    "2019-datacenter-zhcn-g2",
    "2022-datacenter",
    "2022-datacenter-azure-edition",
    "2022-datacenter-azure-edition-core",
    "2022-datacenter-azure-edition-core-smalldisk",
    "2022-datacenter-azure-edition-smalldisk",
    "2022-datacenter-core",
    "2022-datacenter-core-g2",
    "2022-datacenter-core-smalldisk",
    "2022-datacenter-core-smalldisk-g2",
    "2022-datacenter-g2",
    "2022-datacenter-smalldisk",
    "2022-datacenter-smalldisk-g2"
],
"metadata": {
    "description": "The Windows version for the VM. This will pick a fully patched image of this given Windows version."
},
"vmSize": {
    "type": "string",
    "defaultValue": "Standard_D2s_v5",
    "metadata": {
        "description": "Size of the virtual machine."
    }
},
"location": {
    "type": "string",
    "defaultValue": "[resourceGroup().location]",
    "metadata": {
        "description": "Location for all resources."
    }
},
"vmName": {
    "type": "string",
    "defaultValue": "simple-vm",
    "metadata": {
        "description": "Name of the virtual machine."
    }
},
"variables": {
    "storageAccountName": "[format('bootdiags{0}', uniqueString(resourceGroup().id))]",
    "nicName": "myVMNic",
    "addressPrefix": "10.0.0.0/16",
    "subnetName": "Subnet",
    "subnetPrefix": "10.0.0.0/24",
    "virtualNetworkName": "MyVNET",
    "networkSecurityGroupName": "default-NSG"
},
"resources": [
{
    "type": "Microsoft.Storage/storageAccounts",
    "apiVersion": "2021-04-01",
    "name": "[variables('storageAccountName')]",
    "location": "[parameters('location')]"
}
]
}

```

```

    "sku": {
      "name": "Standard_LRS"
    },
    "kind": "Storage"
  },
  {
    "type": "Microsoft.Network/publicIPAddresses",
    "apiVersion": "2021-02-01",
    "name": "[parameters('publicIpName')]",
    "location": "[parameters('location')]",
    "sku": {
      "name": "[parameters('publicIpSku')]"
    },
    "properties": {
      "publicIPAllocationMethod": "[parameters('publicIPAllocationMethod')]",
      "dnsSettings": {
        "domainNameLabel": "[parameters('dnsLabelPrefix')]"
      }
    }
  },
  {
    "type": "Microsoft.Network/networkSecurityGroups",
    "apiVersion": "2021-02-01",
    "name": "[variables('networkSecurityGroupName')]",
    "location": "[parameters('location')]",
    "properties": {
      "securityRules": [
        {
          "name": "default-allow-3389",
          "properties": {
            "priority": 1000,
            "access": "Allow",
            "direction": "Inbound",
            "destinationPortRange": "3389",
            "protocol": "Tcp",
            "sourcePortRange": "*",
            "sourceAddressPrefix": "*",
            "destinationAddressPrefix": "*"
          }
        }
      ]
    }
  },
  {
    "type": "Microsoft.Network/virtualNetworks",
    "apiVersion": "2021-02-01",
    "name": "[variables('virtualNetworkName')]",
    "location": "[parameters('location')]",
    "properties": {
      "addressSpace": {
        "addressPrefixes": [
          "[variables('addressPrefix')]"
        ]
      },
      "subnets": [
        {
          "name": "[variables('subnetName')]",
          "properties": {
            "addressPrefix": "[variables('subnetPrefix')]",
            "networkSecurityGroup": {
              "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('networkSecurityGroupName'))]"
            }
          }
        }
      ]
    },
    "dependsOn": [
      "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"
    ]
  }
}

```

```

        ],
      },
      {
        "type": "Microsoft.Network/networkInterfaces",
        "apiVersion": "2021-02-01",
        "name": "[variables('nicName')]",
        "location": "[parameters('location')]",
        "properties": {
          "ipConfigurations": [
            {
              "name": "ipconfig1",
              "properties": {
                "privateIPAllocationMethod": "Dynamic",
                "publicIPAddress": {
                  "id": "[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]"
                },
                "subnet": {
                  "id": "[resourceId('Microsoft.Network/virtualNetworks/subnets',
variables('virtualNetworkName'), variables('subnetName'))]"
                }
              }
            }
          ]
        },
        "dependsOn": [
          "[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]",
          "[resourceId('Microsoft.Network/virtualNetworks', variables('virtualNetworkName'))]"
        ]
      },
      {
        "type": "Microsoft.Compute/virtualMachines",
        "apiVersion": "2021-03-01",
        "name": "[parameters('vmName')]",
        "location": "[parameters('location')]",
        "properties": {
          "hardwareProfile": {
            "vmSize": "[parameters('vmSize')]"
          },
          "osProfile": {
            "computerName": "[parameters('vmName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "adminPassword": "[parameters('adminPassword')]"
          },
          "storageProfile": {
            "imageReference": {
              "publisher": "MicrosoftWindowsServer",
              "offer": "WindowsServer",
              "sku": "[parameters('OSVersion')]",
              "version": "latest"
            },
            "osDisk": {
              "createOption": "FromImage",
              "managedDisk": {
                "storageAccountType": "StandardSSD_LRS"
              }
            }
          },
          "dataDisks": [
            {
              "diskSizeGB": 1023,
              "lun": 0,
              "createOption": "Empty"
            }
          ]
        },
        "networkProfile": {
          "networkInterfaces": [
            {
              "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]"
            }
          ]
        }
      }
    ]
  }
}

```

```

        ],
        "diagnosticsProfile": {
            "bootDiagnostics": {
                "enabled": true,
                "storageUri": "[reference(resourceId('Microsoft.Storage/storageAccounts',
variables('storageAccountName'))).primaryEndpoints.blob]"
            }
        }
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]",
        "[resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))]"
    ]
},
"outputs": {
    "hostname": {
        "type": "string",
        "value": "[reference(resourceId('Microsoft.Network/publicIPAddresses',
parameters('publicIpName'))).dnsSettings.fqdn]"
    }
}
}

```

Several resources are defined in the template:

- **Microsoft.Network/virtualNetworks/subnets**: create a subnet.
- **Microsoft.Storage/storageAccounts**: create a storage account.
- **Microsoft.Network/publicIPAddresses**: create a public IP address.
- **Microsoft.Network/networkSecurityGroups**: create a network security group.
- **Microsoft.Network/virtualNetworks**: create a virtual network.
- **Microsoft.Network/networkInterfaces**: create a NIC.
- **Microsoft.Compute/virtualMachines**: create a virtual machine.

## Deploy the template

1. Select the following image to sign in to Azure and open a template. The template creates a key vault and a secret.



2. Select or enter the following values. Use the default values, when available.

- **Subscription**: select an Azure subscription.
- **Resource group**: select an existing resource group from the drop-down, or select **Create new**, enter a unique name for the resource group, and then click **OK**.
- **Location**: select a location. For example, **Central US**.
- **Admin username**: provide a username, such as *azureuser*.
- **Admin password**: provide a password to use for the admin account. The password must be at least 12 characters long and meet the **defined complexity requirements**.
- **DNS label prefix**: enter a unique identifier to use as part of the DNS label.
- **Windows OS version**: select which version of Windows you want to run on the VM.
- **VM size**: select the **size** to use for the VM.
- **Location**: the default is the same location as the resource group, if it already exists.

3. Select **Review + create**. After validation completes, select **Create** to create and deploy the VM.

The Azure portal is used to deploy the template. In addition to the Azure portal, you can also use the Azure PowerShell, Azure CLI, and REST API. To learn other deployment methods, see [Deploy templates](#).

## Review deployed resources

You can use the Azure portal to check on the VM and other resource that were created. After the deployment is finished, select **Go to resource group** to see the VM and other resources.

## Clean up resources

When no longer needed, delete the resource group, which deletes the VM and all of the resources in the resource group.

1. Select the **Resource group**.
2. On the page for the resource group, select **Delete**.
3. When prompted, type the name of the resource group and then select **Delete**.

## Next steps

In this quickstart, you deployed a simple virtual machine using an ARM template. To learn more about Azure virtual machines, continue to the tutorial for Linux VMs.

[Azure Windows virtual machine tutorials](#)

# Tutorial: Create and Manage Linux VMs with the Azure CLI

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure virtual machines provide a fully configurable and flexible computing environment. This tutorial covers basic Azure virtual machine deployment items such as selecting a VM size, selecting a VM image, and deploying a VM. You learn how to:

- Create and connect to a VM
- Select and use VM images
- View and use specific VM sizes
- Resize a VM
- View and understand VM state

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create resource group

Create a resource group with the `az group create` command.

An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine. In this example, a resource group named `myResourceGroupVM` is created in the `eastus` region.

```
az group create --name myResourceGroupVM --location eastus
```

The resource group is specified when creating or modifying a VM, which can be seen throughout this tutorial.

## Create virtual machine

Create a virtual machine with the `az vm create` command.

When you create a virtual machine, several options are available such as operating system image, disk sizing, and administrative credentials. The following example creates a VM named `myVM` that runs Ubuntu Server. A user account named `azureuser` is created on the VM, and SSH keys are generated if they do not exist in the default key location (`~/.ssh`):

```
az vm create \
  --resource-group myResourceGroupVM \
  --name myVM \
  --image UbuntuLTS \
  --admin-username azureuser \
  --generate-ssh-keys
```

It may take a few minutes to create the VM. Once the VM has been created, the Azure CLI outputs information about the VM. Take note of the `publicIpAddress`, this address can be used to access the virtual machine..

```
{  
    "fqdns": "",  
    "id": "/subscriptions/d5b9d4b7-6fc1-0000-0000-  
000000000000/resourceGroups/myResourceGroupVM/providers/Microsoft.Compute/virtualMachines/myVM",  
    "location": "eastus",  
    "macAddress": "00-0D-3A-23-9A-49",  
    "powerState": "VM running",  
    "privateIpAddress": "10.0.0.4",  
    "publicIpAddress": "52.174.34.95",  
    "resourceGroup": "myResourceGroupVM"  
}
```

## Connect to VM

You can now connect to the VM with SSH in the Azure Cloud Shell or from your local computer. Replace the example IP address with the `publicIpAddress` noted in the previous step.

```
ssh azureuser@52.174.34.95
```

Once logged in to the VM, you can install and configure applications. When you are finished, you close the SSH session as normal:

```
exit
```

## Understand VM images

The Azure marketplace includes many images that can be used to create VMs. In the previous steps, a virtual machine was created using an Ubuntu image. In this step, the Azure CLI is used to search the marketplace for a CentOS image, which is then used to deploy a second virtual machine.

To see a list of the most commonly used images, use the [az vm image list](#) command.

```
az vm image list --output table
```

The command output returns the most popular VM images on Azure.

Offer	Publisher	Sku	Urn
UrnAlias	Version		
WindowsServer	MicrosoftWindowsServer	2016-Datacenter	MicrosoftWindowsServer:WindowsServer:2016-Datacenter:latest
	Win2016Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2012-R2-Datacenter	MicrosoftWindowsServer:WindowsServer:2012-R2-Datacenter:latest
	Win2012R2Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2008-R2-SP1	MicrosoftWindowsServer:WindowsServer:2008-R2-SP1:latest
	Win2008R2SP1	latest	
WindowsServer	MicrosoftWindowsServer	2012-Datacenter	MicrosoftWindowsServer:WindowsServer:2012-Datacenter:latest
	Win2012Datacenter	latest	
UbuntuServer	Canonical	16.04-LTS	Canonical:UbuntuServer:16.04-LTS:latest
UbuntuLTS		latest	
CentOS	OpenLogic	7.3	OpenLogic:CentOS:7.3:latest
CentOS		latest	
openSUSE-Leap	SUSE	42.2	SUSE:openSUSE-Leap:42.2:latest
openSUSE-Leap		latest	
RHEL	RedHat	7.3	RedHat:RHEL:7.3:latest
RHEL		latest	
SLES	SUSE	12-SP2	SUSE:SLES:12-SP2:latest
SLES		latest	
Debian	credativ	8	credativ:Debian:8:latest
Debian		latest	
CoreOS	CoreOS	Stable	CoreOS:CoreOS:Stable:latest
CoreOS		latest	

A full list can be seen by adding the `--all` parameter. The image list can also be filtered by `--publisher` or `--offer`. In this example, the list is filtered for all images with an offer that matches *CentOS*.

```
az vm image list --offer CentOS --all --output table
```

Partial output:

Offer	Publisher	Sku	Urn	Version
CentOS	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.201501	6.5.201501
Centos	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.201503	6.5.201503
Centos	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.201506	6.5.201506
Centos	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.20150904	6.5.20150904
Centos	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.20160309	6.5.20160309
CentOS	OpenLogic	6.5	OpenLogic:CentOS:6.5:6.5.20170207	6.5.20170207

To deploy a VM using a specific image, take note of the value in the *Urn* column, which consists of the publisher, offer, SKU, and optionally a version number to [identify](#) the image. When specifying the image, the image version number can be replaced with `latest`, which selects the latest version of the distribution. In this example, the `--image` parameter is used to specify the latest version of a CentOS 6.5 image.

```
az vm create --resource-group myResourceGroupVM --name myVM2 --image OpenLogic:CentOS:6.5:latest --generate-ssh-keys
```

## Understand VM sizes

A virtual machine size determines the amount of compute resources such as CPU, GPU, and memory that are made available to the virtual machine. Virtual machines need to be sized appropriately for the expected work load. If workload increases, an existing virtual machine can be resized.

### VM Sizes

The following table categorizes sizes into use cases.

TYPE	COMMON SIZES	DESCRIPTION
General purpose	B, Dsv3, Dv3, DSv2, Dv2, Av2, DC	Balanced CPU-to-memory. Ideal for dev / test and small to medium applications and data solutions.
Compute optimized	Fsv2	High CPU-to-memory. Good for medium traffic applications, network appliances, and batch processes.
Memory optimized	Esv3, Ev3, M, DSv2, Dv2	High memory-to-core. Great for relational databases, medium to large caches, and in-memory analytics.
Storage optimized	Lsv2, Ls	High disk throughput and IO. Ideal for Big Data, SQL, and NoSQL databases.
GPU	NV, NVv2, NC, NCv2, NCv3, ND	Specialized VMs targeted for heavy graphic rendering and video editing.
High performance	H	Our most powerful CPU VMs with optional high-throughput network interfaces (RDMA).

## Find available VM sizes

To see a list of VM sizes available in a particular region, use the [az vm list-sizes](#) command.

```
az vm list-sizes --location eastus --output table
```

Partial output:

ResourceDiskSizeInMb	MaxDataDiskCount	MemoryInMb	Name	NumberofCores	OsDiskSizeInMb
7168	2	3584	Standard_DS1	1	1047552
14336	4	7168	Standard_DS2	2	1047552
28672	8	14336	Standard_DS3	4	1047552
57344	16	28672	Standard_DS4	8	1047552
28672	4	14336	Standard_DS11	2	1047552
57344	8	28672	Standard_DS12	4	1047552
114688	16	57344	Standard_DS13	8	1047552
229376	32	114688	Standard_DS14	16	1047552
20480	1	768	Standard_A0	1	1047552
71680	2	1792	Standard_A1	1	1047552
138240	4	3584	Standard_A2	2	1047552
291840	8	7168	Standard_A3	4	1047552
138240	4	14336	Standard_A5	2	1047552
619520	16	14336	Standard_A4	8	1047552
291840	8	28672	Standard_A6	4	1047552
619520	16	57344	Standard_A7	8	1047552

## Create VM with specific size

In the previous VM creation example, a size was not provided, which results in a default size. A VM size can be selected at creation time using `az vm create` and the `--size` parameter.

```
az vm create \
    --resource-group myResourceGroupVM \
    --name myVM3 \
    --image UbuntuLTS \
    --size Standard_F4s \
    --generate-ssh-keys
```

## Resize a VM

After a VM has been deployed, it can be resized to increase or decrease resource allocation. You can view the current size of a VM with `az vm show`:

```
az vm show --resource-group myResourceGroupVM --name myVM --query hardwareProfile.vmSize
```

Before resizing a VM, check if the desired size is available on the current Azure cluster. The `az vm list-vm-resize-options` command returns the list of sizes.

```
az vm list-vm-resize-options --resource-group myResourceGroupVM --name myVM --query [].name
```

If the desired size is available, the VM can be resized from a powered-on state, however it is rebooted during the operation. Use the [az vm resize](#) command to perform the resize.

```
az vm resize --resource-group myResourceGroupVM --name myVM --size Standard_DS4_v2
```

If the desired size is not on the current cluster, the VM needs to be deallocated before the resize operation can occur. Use the [az vm deallocate](#) command to stop and deallocate the VM. Note, when the VM is powered back on, any data on the temp disk may be removed. The public IP address also changes unless a static IP address is being used.

```
az vm deallocate --resource-group myResourceGroupVM --name myVM
```

Once deallocated, the resize can occur.

```
az vm resize --resource-group myResourceGroupVM --name myVM --size Standard_GS1
```

After the resize, the VM can be started.

```
az vm start --resource-group myResourceGroupVM --name myVM
```

## VM power states

An Azure VM can have one of many power states. This state represents the current state of the VM from the standpoint of the hypervisor.

### Power states

POWER STATE	DESCRIPTION
Starting	Indicates the virtual machine is being started.
Running	Indicates that the virtual machine is running.
Stopping	Indicates that the virtual machine is being stopped.
Stopped	Indicates that the virtual machine is stopped. Virtual machines in the stopped state still incur compute charges.
Deallocating	Indicates that the virtual machine is being deallocated.
Deallocated	Indicates that the virtual machine is removed from the hypervisor but still available in the control plane. Virtual machines in the Deallocated state do not incur compute charges.
-	Indicates that the power state of the virtual machine is unknown.

### Find the power state

To retrieve the state of a particular VM, use the [az vm get-instance-view](#) command. Be sure to specify a valid name for a virtual machine and resource group.

```
az vm get-instance-view \
--name myVM \
--resource-group myResourceGroupVM \
--query instanceView.statuses[1] --output table
```

Output:

Code	DisplayStatus	Level
PowerState/running	VM running	Info

To retrieve the power state of all the VMs in your subscription, use the [Virtual Machines - List All API](#) with parameter `statusOnly` set to *true*.

## Management tasks

During the life-cycle of a virtual machine, you may want to run management tasks such as starting, stopping, or deleting a virtual machine. Additionally, you may want to create scripts to automate repetitive or complex tasks. Using the Azure CLI, many common management tasks can be run from the command line or in scripts.

### Get IP address

This command returns the private and public IP addresses of a virtual machine.

```
az vm list-ip-addresses --resource-group myResourceGroupVM --name myVM --output table
```

### Stop virtual machine

```
az vm stop --resource-group myResourceGroupVM --name myVM
```

### Start virtual machine

```
az vm start --resource-group myResourceGroupVM --name myVM
```

### Deleting VM resources

You can delete a VM, but by default this only deletes the VM resource, not the disks and networking resources the VM uses. You can change the default behavior to delete other resources when you delete the VM. For more information, see [Delete a VM and attached resources](#).

Deleting a resource group also deletes all resources contained within, such as the VM, virtual network, and disk. The `--no-wait` parameter returns control to the prompt without waiting for the operation to complete. The `--yes` parameter confirms that you wish to delete the resources without an additional prompt to do so.

```
az group delete --name myResourceGroupVM --no-wait --yes
```

## Next steps

In this tutorial, you learned about basic VM creation and management such as how to:

- Create and connect to a VM
- Select and use VM images
- View and use specific VM sizes

- Resize a VM
- View and understand VM state

Advance to the next tutorial to learn about VM disks.

[Create and Manage VM disks](#)

# Tutorial - Manage Azure disks with the Azure CLI

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure virtual machines (VMs) use disks to store the operating system, applications, and data. When you create a VM, it is important to choose a disk size and configuration appropriate to the expected workload. This tutorial shows you how to deploy and manage VM disks. You learn about:

- OS disks and temporary disks
- Data disks
- Standard and Premium disks
- Disk performance
- Attaching and preparing data disks
- Disk snapshots

## Default Azure disks

When an Azure virtual machine is created, two disks are automatically attached to the virtual machine.

**Operating system disk** - Operating system disks can be sized up to 2 TB, and hosts the VMs operating system. The OS disk is labeled `/dev/sda` by default. The disk caching configuration of the OS disk is optimized for OS performance. Because of this configuration, the OS disk **should not** be used for applications or data. For applications and data, use data disks, which are detailed later in this tutorial.

**Temporary disk** - Temporary disks use a solid-state drive that is located on the same Azure host as the VM. Temp disks are highly performant and may be used for operations such as temporary data processing. However, if the VM is moved to a new host, any data stored on a temporary disk is removed. The size of the temporary disk is determined by the VM size. Temporary disks are labeled `/dev/sdb` and have a mountpoint of `/mnt`.

## Azure data disks

To install applications and store data, additional data disks can be added. Data disks should be used in any situation where durable and responsive data storage is desired. The size of the virtual machine determines how many data disks can be attached to a VM.

## VM disk types

Azure provides two types of disks.

**Standard disks** - backed by HDDs, and delivers cost-effective storage while still being performant. Standard disks are ideal for a cost effective dev and test workload.

**Premium disks** - backed by SSD-based, high-performance, low-latency disk. Perfect for VMs running production workload. VM sizes with an S in the [size name](#), typically support Premium Storage. For example, DS-series, DSv2-series, GS-series, and FS-series VMs support premium storage. When you select a disk size, the value is rounded up to the next type. For example, if the disk size is more than 64 GB, but less than 128 GB, the disk type is P10.



PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Eligi ble for rese rvat ion	No	No	No	No	No	No	No	No	Yes, up to one yea r	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year

\*Applies only to disks with on-demand bursting enabled.

When you provision a premium storage disk, unlike standard storage, you are guaranteed the capacity, IOPS, and throughput of that disk. For example, if you create a P50 disk, Azure provisions 4,095-GB storage capacity, 7,500 IOPS, and 250-MB/s throughput for that disk. Your application can use all or part of the capacity and performance. Premium SSD disks are designed to provide low single-digit millisecond latencies and target IOPS and throughput described in the preceding table 99.9% of the time.

While the above table identifies max IOPS per disk, a higher level of performance can be achieved by striping multiple data disks. For instance, 64 data disks can be attached to Standard\_GS5 VM. If each of these disks is sized as a P30, a maximum of 80,000 IOPS can be achieved. For detailed information on max IOPS per VM, see [VM types and sizes](#).

## Launch Azure Cloud Shell

Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open Cloud Shell, select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create and attach disks

Data disks can be created and attached at VM creation time or to an existing VM.

### Attach disk at VM creation

Create a resource group with the [az group create](#) command.

```
az group create --name myResourceGroupDisk --location eastus
```

Create a VM using the [az vm create](#) command. The following example creates a VM named *myVM*, adds a user account named *azureuser*, and generates SSH keys if they do not exist. The `--datadisk-sizes-gb` argument is used to specify that an additional disk should be created and attached to the virtual machine. To create and attach more than one disk, use a space-delimited list of disk size values. In the following example, a VM is created with two data disks, both 128 GB. Because the disk sizes are 128 GB, these disks are both configured as P10s, which provide maximum 500 IOPS per disk.

```
az vm create \
--resource-group myResourceGroupDisk \
--name myVM \
--image UbuntuLTS \
--size Standard_DS2_v2 \
--admin-username azureuser \
--generate-ssh-keys \
--data-disk-sizes-gb 128 128
```

## Attach disk to existing VM

To create and attach a new disk to an existing virtual machine, use the [az vm disk attach](#) command. The following example creates a premium disk, 128 gigabytes in size, and attaches it to the VM created in the last step.

```
az vm disk attach \
--resource-group myResourceGroupDisk \
--vm-name myVM \
--name myDataDisk \
--size-gb 128 \
--sku Premium_LRS \
--new
```

## Prepare data disks

Once a disk has been attached to the virtual machine, the operating system needs to be configured to use the disk. The following example shows how to manually configure a disk. This process can also be automated using cloud-init, which is covered in a [later tutorial](#).

Create an SSH connection with the virtual machine. Replace the example IP address with the public IP of the virtual machine.

```
ssh azureuser@10.101.10.10
```

Partition the disk with `parted`.

```
sudo parted /dev/sdc --script mklabel gpt mkpart xfspart xfs 0% 100%
```

Write a file system to the partition by using the `mkfs` command. Use `partprobe` to make the OS aware of the change.

```
sudo mkfs.xfs /dev/sdc1
sudo partprobe /dev/sdc1
```

Mount the new disk so that it is accessible in the operating system.

```
sudo mkdir /datadrive && sudo mount /dev/sdc1 /datadrive
```

The disk can now be accessed through the `/datadrive` mountpoint, which can be verified by running the `df -h` command.

```
df -h | grep -i "sd"
```

The output shows the new drive mounted on `/datadrive`.

Filesystem	Size	Used	Avail	Use%	Mounted on
<code>/dev/sda1</code>	29G	2.0G	27G	7%	/
<code>/dev/sda15</code>	105M	3.6M	101M	4%	/boot/efi
<code>/dev/sdb1</code>	14G	41M	13G	1%	/mnt
<code>/dev/sdc1</code>	50G	52M	47G	1%	/datadrive

To ensure that the drive is remounted after a reboot, it must be added to the `/etc/fstab` file. To do so, get the UUID of the disk with the `blkid` utility.

```
sudo -i blkid
```

The output displays the UUID of the drive, `/dev/sdc1` in this case.

```
/dev/sdc1: UUID="33333333-3b3b-3c3c-3d3d-3e3e3e3e3e" TYPE="xfs"
```

#### NOTE

Improperly editing the `/etc/fstab` file could result in an unbootable system. If unsure, refer to the distribution's documentation for information on how to properly edit this file. It is also recommended that a backup of the `/etc/fstab` file is created before editing.

Open the `/etc/fstab` file in a text editor as follows:

```
sudo nano /etc/fstab
```

Add a line similar to the following to the `/etc/fstab` file, replacing the UUID value with your own.

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e /datadrive xfs defaults,nofail 1 2
```

When you are done editing the file, use `Ctrl+O` to write the file and `Ctrl+X` to exit the editor.

Now that the disk has been configured, close the SSH session.

```
exit
```

## Take a disk snapshot

When you take a disk snapshot, Azure creates a read only, point-in-time copy of the disk. Azure VM snapshots are useful to quickly save the state of a VM before you make configuration changes. In the event of an issue or error, VM can be restored using a snapshot. When a VM has more than one disk, a snapshot is taken of each disk independently of the others. To take application consistent backups, consider stopping the VM before you take disk snapshots. Alternatively, use the [Azure Backup service](#), which enables you to perform automated backups while the VM is running.

### Create snapshot

Before you create a snapshot, you need the ID or name of the disk. Use `az vm show` to show the disk ID. In this example, the disk ID is stored in a variable so that it can be used in a later step.

```
osdiskid=$(az vm show \
-g myResourceGroupDisk \
-n myVM \
--query "storageProfile.osDisk.managedDisk.id" \
-o tsv)
```

Now that you have the ID, use [az snapshot create](#) to create a snapshot of the disk.

```
az snapshot create \
--resource-group myResourceGroupDisk \
--source "$osdiskid" \
--name osDisk-backup
```

## Create disk from snapshot

This snapshot can then be converted into a disk using [az disk create](#), which can be used to recreate the virtual machine.

```
az disk create \
--resource-group myResourceGroupDisk \
--name mySnapshotDisk \
--source osDisk-backup
```

## Restore virtual machine from snapshot

To demonstrate virtual machine recovery, delete the existing virtual machine using [az vm delete](#).

```
az vm delete \
--resource-group myResourceGroupDisk \
--name myVM
```

Create a new virtual machine from the snapshot disk.

```
az vm create \
--resource-group myResourceGroupDisk \
--name myVM \
--attach-os-disk mySnapshotDisk \
--os-type linux
```

## Reattach data disk

All data disks need to be reattached to the virtual machine.

Find the data disk name using the [az disk list](#) command. This example places the name of the disk in a variable named `datadisk`, which is used in the next step.

```
datadisk=$(az disk list \
-g myResourceGroupDisk \
--query "[?contains(name,'myVM')].[id]" \
-o tsv)
```

Use the [az vm disk attach](#) command to attach the disk.

```
az vm disk attach \
-g myResourceGroupDisk \
--vm-name myVM \
--name $datadisk
```

## Next steps

In this tutorial, you learned about VM disks topics such as:

- OS disks and temporary disks
- Data disks
- Standard and Premium disks
- Disk performance
- Attaching and preparing data disks
- Disk snapshots

Advance to the next tutorial to learn about automating VM configuration.

[Automate VM configuration](#)

# Tutorial - How to use cloud-init to customize a Linux virtual machine in Azure on first boot

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In a previous tutorial, you learned how to SSH to a virtual machine (VM) and manually install NGINX. To create VMs in a quick and consistent manner, some form of automation is typically desired. A common approach to customize a VM on first boot is to use [cloud-init](#). In this tutorial you learn how to:

- Create a cloud-init config file
- Create a VM that uses a cloud-init file
- View a running Node.js app after the VM is created
- Use Key Vault to securely store certificates
- Automate secure deployments of NGINX with cloud-init

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Cloud-init overview

[Cloud-init](#) is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files, or to configure users and security. As cloud-init runs during the initial boot process, there are no additional steps or required agents to apply your configuration.

Cloud-init also works across distributions. For example, you don't use `apt-get install` or `yum install` to install a package. Instead you can define a list of packages to install. Cloud-init automatically uses the native package management tool for the distro you select.

We are working with our partners to get cloud-init included and working in the images that they provide to Azure. For detailed information cloud-init support for each distribution, see [Cloud-init support for VMs in Azure](#).

## Create cloud-init config file

To see cloud-init in action, create a VM that installs NGINX and runs a simple 'Hello World' Node.js app. The following cloud-init configuration installs the required packages, creates a Node.js app, then initialize and starts the app.

At your bash prompt or in the Cloud Shell, create a file named `cloud-init.txt` and paste the following configuration. For example, type `sensible-editor cloud-init.txt` to create the file and see a list of available editors. Make sure that the whole cloud-init file is copied correctly, especially the first line:

```

#cloud-config
package_upgrade: true
packages:
- nginx
- nodejs
- npm
write_files:
- owner: www-data:www-data
  path: /etc/nginx/sites-available/default
  content: |
    server {
      listen 80;
      location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection keep-alive;
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
      }
    }
- owner: azureuser:azureuser
  path: /home/azureuser/myapp/index.js
  content: |
    var express = require('express')
    var app = express()
    var os = require('os');
    app.get('/', function (req, res) {
      res.send('Hello World from host ' + os.hostname() + '!')
    })
    app.listen(3000, function () {
      console.log('Hello world app listening on port 3000!')
    })
runcmd:
- service nginx restart
- cd "/home/azureuser/myapp"
- npm init
- npm install express -y
- nodejs index.js

```

For more information about cloud-init configuration options, see [cloud-init config examples](#).

## Create virtual machine

Before you can create a VM, create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroupAutomate* in the *eastus* location:

```
az group create --name myResourceGroupAutomate --location eastus
```

Now create a VM with [az vm create](#). Use the `--custom-data` parameter to pass in your cloud-init config file. Provide the full path to the *cloud-init.txt* config if you saved the file outside of your present working directory. The following example creates a VM named *myVM*:

```
az vm create \
--resource-group myResourceGroupAutomate \
--name myAutomatedVM \
--image UbuntuLTS \
--admin-username azureuser \
--generate-ssh-keys \
--custom-data cloud-init.txt
```

It takes a few minutes for the VM to be created, the packages to install, and the app to start. There are background tasks that continue to run after the Azure CLI returns you to the prompt. It may be another couple of minutes before you can access the app. When the VM has been created, take note of the `publicIpAddress` displayed by the Azure CLI. This address is used to access the Node.js app via a web browser.

To allow web traffic to reach your VM, open port 80 from the Internet with [az vm open-port](#):

```
az vm open-port --port 80 --resource-group myResourceGroupAutomate --name myAutomatedVM
```

## Test web app

Now you can open a web browser and enter `http://<publicIpAddress>` in the address bar. Provide your own public IP address from the VM create process. Your Node.js app is displayed as shown in the following example:



## Inject certificates from Key Vault

This optional section shows how you can securely store certificates in Azure Key Vault and inject them during the VM deployment. Rather than using a custom image that includes the certificates baked-in, this process ensures that the most up-to-date certificates are injected to a VM on first boot. During the process, the certificate never leaves the Azure platform or is exposed in a script, command-line history, or template.

Azure Key Vault safeguards cryptographic keys and secrets, such as certificates or passwords. Key Vault helps streamline the key management process and enables you to maintain control of keys that access and encrypt your data. This scenario introduces some Key Vault concepts to create and use a certificate, though is not an exhaustive overview on how to use Key Vault.

The following steps show how you can:

- Create an Azure Key Vault
- Generate or upload a certificate to the Key Vault
- Create a secret from the certificate to inject in to a VM
- Create a VM and inject the certificate

### Create an Azure Key Vault

First, create a Key Vault with [az keyvault create](#) and enable it for use when you deploy a VM. Each Key Vault requires a unique name, and should be all lower case. Replace `mykeyvault` in the following example with your own unique Key Vault name:

```
keyvault_name=mykeyvault
az keyvault create \
    --resource-group myResourceGroupAutomate \
    --name $keyvault_name \
    --enabled-for-deployment
```

### Generate certificate and store in Key Vault

For production use, you should import a valid certificate signed by trusted provider with [az keyvault certificate import](#). For this tutorial, the following example shows how you can generate a self-signed certificate with [az keyvault certificate create](#) that uses the default certificate policy:

```
az keyvault certificate create \
--vault-name $keyvault_name \
--name mycert \
--policy "$(az keyvault certificate get-default-policy --output json)"
```

## Prepare certificate for use with VM

To use the certificate during the VM create process, obtain the ID of your certificate with [az keyvault secret list-versions](#). The VM needs the certificate in a certain format to inject it on boot, so convert the certificate with [az vm secret format](#). The following example assigns the output of these commands to variables for ease of use in the next steps:

```
secret=$(az keyvault secret list-versions \
--vault-name $keyvault_name \
--name mycert \
--query "[?attributes.enabled].id" --output tsv)
vm_secret=$(az vm secret format --secret "$secret" --output json)
```

## Create cloud-init config to secure NGINX

When you create a VM, certificates and keys are stored in the protected `/var/lib/waagent`/directory. To automate adding the certificate to the VM and configuring NGINX, you can use an updated cloud-init config from the previous example.

Create a file named `cloud-init-secured.txt` and paste the following configuration. If you use the Cloud Shell, create the cloud-init config file there and not on your local machine. For example, type

```
sensible-editor cloud-init-secured.txt
```

 to create the file and see a list of available editors. Make sure that the whole cloud-init file is copied correctly, especially the first line:

```

#cloud-config
package_upgrade: true
packages:
- nginx
- nodejs
- npm
write_files:
- owner: www-data:www-data
  path: /etc/nginx/sites-available/default
  content: |
    server {
      listen 80;
      listen 443 ssl;
      ssl_certificate /etc/nginx/ssl/mycert.cert;
      ssl_certificate_key /etc/nginx/ssl/mycert.prv;
      location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection keep-alive;
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
      }
    }
- owner: azureuser:azureuser
  path: /home/azureuser/myapp/index.js
  content: |
    var express = require('express')
    var app = express()
    var os = require('os');
    app.get('/', function (req, res) {
      res.send('Hello World from host ' + os.hostname() + '!')
    })
    app.listen(3000, function () {
      console.log('Hello world app listening on port 3000!')
    })
runcmd:
- secretsname=$(find /var/lib/waagent/ -name "*.prv" | cut -c -57)
- mkdir /etc/nginx/ssl
- cp $secretsname.crt /etc/nginx/ssl/mycert.cert
- cp $secretsname.prv /etc/nginx/ssl/mycert.prv
- service nginx restart
- cd "/home/azureuser/myapp"
- npm init
- npm install express -y
- nodejs index.js

```

## Create secure VM

Now create a VM with `az vm create`. The certificate data is injected from Key Vault with the `--secrets` parameter. As in the previous example, you also pass in the cloud-init config with the `--custom-data` parameter:

```

az vm create \
--resource-group myResourceGroupAutomate \
--name myVMWithCerts \
--image UbuntuLTS \
--admin-username azureuser \
--generate-ssh-keys \
--custom-data cloud-init-secured.txt \
--secrets "$vm_secret"

```

It takes a few minutes for the VM to be created, the packages to install, and the app to start. There are background tasks that continue to run after the Azure CLI returns you to the prompt. It may be another couple of minutes before you can access the app. When the VM has been created, take note of the `publicIpAddress`

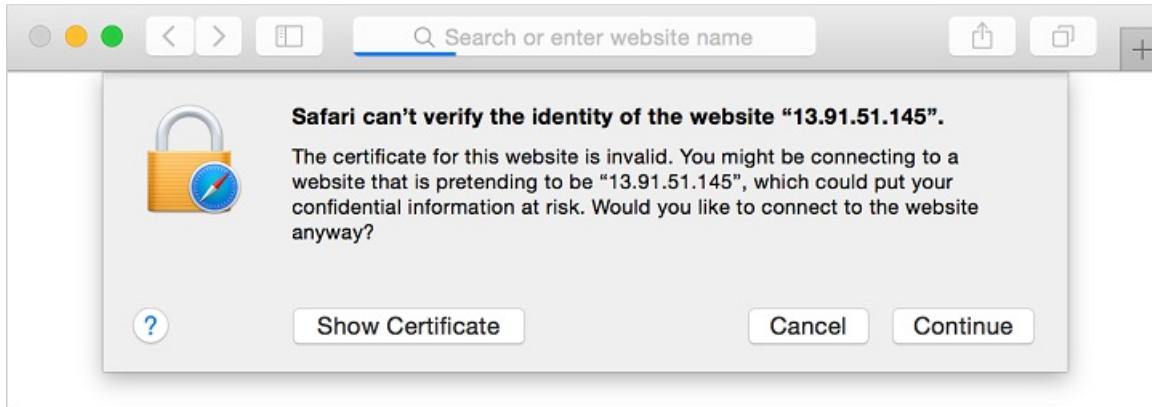
displayed by the Azure CLI. This address is used to access the Node.js app via a web browser.

To allow secure web traffic to reach your VM, open port 443 from the Internet with [az vm open-port](#):

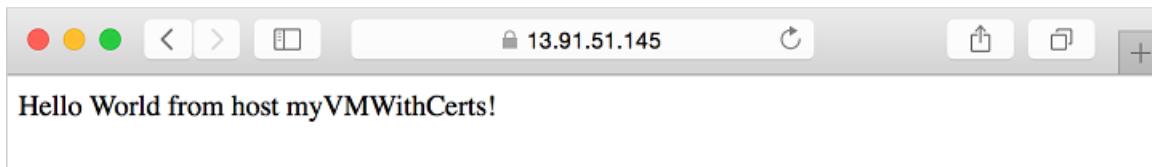
```
az vm open-port \
--resource-group myResourceGroupAutomate \
--name myVMWithCerts \
--port 443
```

### Test secure web app

Now you can open a web browser and enter `https://<publicIpAddress>` in the address bar. Provide your own public IP address as shown in the output of the previous VM create process. Accept the security warning if you used a self-signed certificate:



Your secured NGINX site and Node.js app is then displayed as in the following example:



## Next steps

In this tutorial, you configured VMs on first boot with cloud-init. You learned how to:

- Create a cloud-init config file
- Create a VM that uses a cloud-init file
- View a running Node.js app after the VM is created
- Use Key Vault to securely store certificates
- Automate secure deployments of NGINX with cloud-init

Advance to the next tutorial to learn how to create custom VM images.

[Create custom VM images](#)

# Tutorial: Create a custom image of an Azure VM with the Azure CLI

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Custom images are like marketplace images, but you create them yourself. Custom images can be used to bootstrap configurations such as preloading applications, application configurations, and other OS configurations. In this tutorial, you create your own custom image of an Azure virtual machine. You learn how to:

- Create an Azure Compute Gallery (formerly known as Shared Image Gallery)
- Create an image definition
- Create an image version
- Create a VM from an image
- Share a gallery

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.35.0 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Overview

An [Azure Compute Gallery](#) simplifies custom image sharing across your organization. Custom images are like marketplace images, but you create them yourself. Custom images can be used to bootstrap configurations such as preloading applications, application configurations, and other OS configurations.

The Azure Compute Gallery lets you share your custom VM images with others. Choose which images you want to share, which regions you want to make them available in, and who you want to share them with.

The Azure Compute Gallery feature has multiple resource types:

RESOURCE	DESCRIPTION
<b>Image source</b>	This is a resource that can be used to create an <a href="#">image version</a> in a gallery. An image source can be an existing Azure VM that is either <a href="#">generalized or specialized</a> , a managed image, a snapshot, or an image version in another gallery.
<b>Gallery</b>	Like the Azure Marketplace, a <a href="#">gallery</a> is a repository for managing and sharing images and <a href="#">VM applications</a> , but you control who has access.
<b>Image definition</b>	Image definitions are created within a gallery and carry information about the image and requirements for using it internally. This includes whether the image is Windows or Linux, release notes, and minimum and maximum memory requirements. It is a definition of a type of image.

RESOURCE	DESCRIPTION
Image version	An <b>image version</b> is what you use to create a VM when using a gallery. You can have multiple versions of an image as needed for your environment. Like a managed image, when you use an <b>image version</b> to create a VM, the image version is used to create new disks for the VM. Image versions can be used multiple times.

## Before you begin

The steps below detail how to take an existing VM and turn it into a reusable custom image that you can use to create new VM instances.

To complete the example in this tutorial, you must have an existing virtual machine. If needed, you can see the [CLI quickstart](#) to create a VM to use for this tutorial. When working through the tutorial, replace the resource names where needed.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create a gallery

A gallery is the primary resource used for enabling image sharing.

Allowed characters for gallery name are uppercase or lowercase letters, digits, dots, and periods. The gallery name cannot contain dashes. Gallery names must be unique within your subscription.

Create a gallery using [az sig create](#). The following example creates a resource group named gallery named *myGalleryRG* in *East US*, and a gallery named *myGallery*.

```
az group create --name myGalleryRG --location eastus
az sig create --resource-group myGalleryRG --gallery-name myGallery
```

## Get information about the VM

You can see a list of VMs that are available using [az vm list](#).

```
az vm list --output table
```

Once you know the VM name and what resource group it is in, get the ID of the VM using [az vm get-instance-view](#).

```
az vm get-instance-view -g MyResourceGroup -n MyVm --query id
```

Copy the ID of your VM to use later.

# Create an image definition

Image definitions create a logical grouping for images. They are used to manage information about the image versions that are created within them.

Image definition names can be made up of uppercase or lowercase letters, digits, dots, dashes, and periods.

For more information about the values you can specify for an image definition, see [Image definitions](#).

Create an image definition in the gallery using `az sig image-definition create`.

In this example, the image definition is named `myImageDefinition`, and is for a [specialized](#) Linux OS image.

```
az sig image-definition create \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--publisher myPublisher \
--offer myOffer \
--sku mySKU \
--os-type Linux \
--os-state specialized
```

Copy the ID of the image definition from the output to use later.

## Create the image version

Create an image version from the VM using `az sig image-version create`.

Allowed characters for image version are numbers and periods. Numbers must be within the range of a 32-bit integer. Format: *MajorVersion.MinorVersion.Patch*.

In this example, the version of our image is `1.0.0` and we are going to create 2 replicas in the *West Central US* region, 1 replica in the *South Central US* region and 1 replica in the *East US 2* region using zone-redundant storage. The replication regions must include the region the source VM is located.

Replace the value of `--managed-image` in this example with the ID of your VM from the previous step.

```
az sig image-version create \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--gallery-image-version 1.0.0 \
--target-regions "westcentralus" "southcentralus=1" "eastus=1=standard_zrs" \
--replica-count 2 \
--managed-image "/subscriptions/<SubscriptionID>/resourceGroups/MyResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM"
```

### NOTE

You need to wait for the image version to completely finish being built and replicated before you can use the same managed image to create another image version.

You can also store your image in Premium storage by adding `--storage-account-type premium_lrs`, or [Zone Redundant Storage](#) by adding `--storage-account-type standard_zrs` when you create the image version.

## Create the VM

Create the VM using `az vm create` using the `--specialized` parameter to indicate the image is a specialized image.

Use the image definition ID for `--image` to create the VM from the latest version of the image that is available.

You can also create the VM from a specific version by supplying the image version ID for `--image`.

In this example, we are creating a VM from the latest version of the *myImageDefinition* image.

```
az group create --name myResourceGroup --location eastus
az vm create --resource-group myResourceGroup \
  --name myVM2 \
  --image "/subscriptions/<Subscription
ID>/resourceGroups/myGalleryRG/providers/Microsoft.Compute/galleries/myGallery/images/myImageDefinition" \
  --specialized
```

## Share the gallery

You can share images across subscriptions using Azure role-based access control (Azure RBAC). You can share images at the gallery, image definition or image version level. Any user that has read permissions to an image version, even across subscriptions, will be able to deploy a VM using the image version.

We recommend that you share with other users at the gallery level. To get the object ID of your gallery, use `az sig show`.

```
az sig show \
  --resource-group myGalleryRG \
  --gallery-name myGallery \
  --query id
```

Use the object ID as a scope, along with an email address and `az role assignment create` to give a user access to the Azure Compute Gallery. Replace `<email-address>` and `<gallery ID>` with your own information.

```
az role assignment create \
  --role "Reader" \
  --assignee <email address> \
  --scope <gallery ID>
```

For more information about how to share resources using Azure RBAC, see [Add or remove Azure role assignments using Azure CLI](#).

## Azure Image Builder

Azure also offers a service, built on Packer, [Azure VM Image Builder](#). Simply describe your customizations in a template, and it will handle the image creation.

## Next steps

In this tutorial, you created a custom VM image. You learned how to:

- Create an Azure Compute Gallery
- Create an image definition
- Create an image version
- Create a VM from an image
- Share a gallery

Advance to the next tutorial to learn about highly available virtual machines.

Create highly available VMs

# Create and deploy virtual machines in an availability set using Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

In this tutorial, you learn how to increase the availability and reliability of your Virtual Machine solutions on Azure using a capability called Availability Sets. Availability sets ensure that the VMs you deploy on Azure are distributed across multiple isolated hardware clusters. Doing this ensures that if a hardware or software failure within Azure happens, only a subset of your VMs is impacted and that your overall solution remains available and operational.

In this tutorial, you learn how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create an availability set

You can create an availability set using `az vm availability-set create`. In this example, the number of update and fault domains is set to 2 for the availability set named *myAvailabilitySet* in the *myResourceGroupAvailability* resource group.

First, create a resource group with [az group create](#), then create the availability set:

```
az group create --name myResourceGroupAvailability --location eastus

az vm availability-set create \
    --resource-group myResourceGroupAvailability \
    --name myAvailabilitySet \
    --platform-fault-domain-count 2 \
    --platform-update-domain-count 2
```

Availability Sets allow you to isolate resources across fault domains and update domains. A **fault domain** represents an isolated collection of server + network + storage resources. In the preceding example, the availability set is distributed across at least two fault domains when the VMs are deployed. The availability set is also distributed across two **update domains**. Two update domains ensure that when Azure performs software updates, the VM resources are isolated, preventing all the software that runs on the VM from being updated at the same time.

## Create VMs inside an availability set

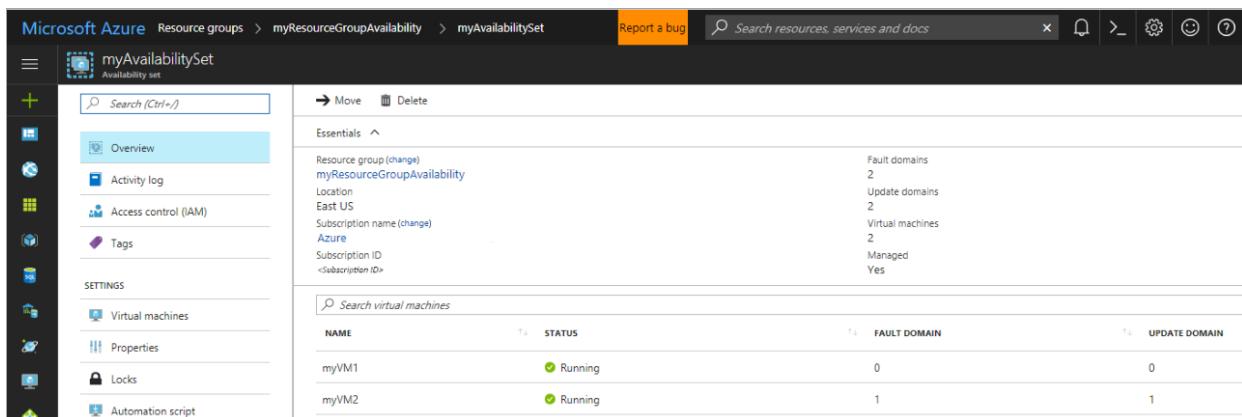
VMs must be created within the availability set to make sure they are correctly distributed across the hardware. An existing VM cannot be added to an availability set after it is created.

When a VM is created with [az vm create](#), use the `--availability-set` parameter to specify the name of the availability set.

```
for i in `seq 1 2`; do
    az vm create \
        --resource-group myResourceGroupAvailability \
        --name myVM$i \
        --availability-set myAvailabilitySet \
        --size Standard_DS1_v2 \
        --vnet-name myVnet \
        --subnet mySubnet \
        --image UbuntuLTS \
        --admin-username azureuser \
        --generate-ssh-keys
done
```

There are now two virtual machines within the availability set. Because they are in the same availability set, Azure ensures that the VMs and all their resources (including data disks) are distributed across isolated physical hardware. This distribution helps ensure much higher availability of the overall VM solution.

The availability set distribution can be viewed in the portal by going to Resource Groups > myResourceGroupAvailability > myAvailabilitySet. The VMs are distributed across the two fault and update domains, as shown in the following example:



NAME	STATUS	FAULT DOMAIN	UPDATE DOMAIN
myVM1	Running	0	0
myVM2	Running	1	1

## Check for available VM sizes

Additional VMs can be added to the availability set later, where VM sizes are available on the hardware. Use [az vm availability-set list-sizes](#) to list all the available sizes on the hardware cluster for the availability set:

```
az vm availability-set list-sizes \
    --resource-group myResourceGroupAvailability \
    --name myAvailabilitySet \
    --output table
```

## Next steps

In this tutorial, you learned how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes

Advance to the next tutorial to learn about virtual machine scale sets.

[Create a virtual machine scale set](#)

- To learn more about availability zones, visit the [Availability Zones documentation](#).
- More documentation about both availability sets and availability zones is also available at [Availability options for Azure Virtual Machines](#).
- To try out availability zones, visit [Create a Linux virtual machine in an availability zone with the Azure CLI](#)

# Tutorial: Create a virtual machine scale set and deploy a highly available app on Linux

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Uniform scale sets

Virtual machine scale sets with [Flexible orchestration](#) let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule.

In this tutorial, you deploy a virtual machine scale set in Azure and learn how to:

- Create a resource group.
- Create a Flexible scale set with a load balancer.
- Add nginx to the scale set instances.
- Open port 80 to HTTP traffic.
- Test the scale set.

## Scale Set overview

Scale sets provide the following key benefits:

- Easy to create and manage multiple VMs
- Provides high availability and application resiliency by distributing VMs across fault domains
- Allows your application to automatically scale as resource demand changes
- Works at large-scale

With Flexible orchestration, Azure provides a unified experience across the Azure VM ecosystem. Flexible orchestration offers high availability guarantees (up to 1000 VMs) by spreading VMs across fault domains in a region or within an Availability Zone. This enables you to scale out your application while maintaining fault domain isolation that is essential to run quorum-based or stateful workloads, including:

- Quorum-based workloads
- Open-source databases
- Stateful applications
- Services that require high availability and large scale
- Services that want to mix virtual machine types or leverage Spot and on-demand VMs together
- Existing Availability Set applications

Learn more about the differences between Uniform scale sets and Flexible scale sets in [Orchestration Modes](#).

## Create a scale set

Use the Azure portal to create a Flexible scale set.

1. Open the [Azure portal](#).
2. Search for and select **Virtual machine scale sets**.
3. Select **Create** on the **Virtual machine scale sets** page. The **Create a virtual machine scale set** will open.
4. Select the subscription that you want to use for **Subscription**.

5. For **Resource group**, select **Create new** and type *myVMSSRG* for the name and then select **OK**.

#### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	myAzureSubscription
Resource group *	(New) myVMSS_group
<a href="#">Create new</a>	

6. For **Virtual machine scale set name**, type *myVMSS*.

7. For **Region**, select a region that is close to you like *East US*.

#### Scale set details

Virtual machine scale set name *	myVMSS
Region *	(US) East US
Availability zone ⓘ	None

8. Leave **Availability zone** as blank for this example.

9. For **Orchestration mode**, select **Flexible**.

10. Leave the default of **1** for fault domain count or choose another value from the drop-down.

#### Orchestration

A scale set has a "scale set model" that defines the attributes of virtual machine instances (size, number of data disks, etc). As the number of instances in the scale set changes, new instances are added based on the scale set model.

[Learn more about the scale set model](#)

Orchestration mode \* ⓘ

- Uniform**: optimized for large scale stateless workloads with identical instances  
 **Flexible (preview)**: achieve high availability at scale with identical or multiple virtual machine types

**!** This virtual machine scale set will be created as a scale set with flexible orchestration mode (preview). To enable the preview, you must register your subscription. [Learn more](#)

Fault domain count \* ⓘ

1

11. For **Image**, select *Ubuntu 18.04 LTS*.

12. For **Size**, leave the default value or select a size like *Standard\_E2s\_V3*.

13. In **Username** type *azureuser*.

14. For **SSH public key source**, leave the default of **Generate new key pair**, and then type *myKey* for the **Key pair name**.

#### Administrator account

Authentication type ⓘ

- Password  
 SSH public key

Username \* ⓘ

azureuser

SSH public key source

Generate new key pair

Key pair name \*

myVMSSKey

15. On the **Networking** tab, under **Load balancing**, select **Use a load balancer**.

16. For **Load balancing options**, leave the default of **Azure load balancer**.

17. For **Select a load balancer**, select **Create new**.

## Load balancing

You can place this virtual machine scale set in the backend pool of an existing Azure load balancing solution. [Learn more ↗](#)

Use a load balancer

### Load balancing settings

- **Application Gateway** is an HTTP/HTTPS web traffic load balancer with URL-based routing, SSL termination, session persistence, and web application firewall. [Learn more about Application Gateway ↗](#)
- **Azure Load Balancer** supports all TCP/UDP network traffic, port-forwarding, and outbound flows. [Learn more about Azure Load Balancer ↗](#)

Load balancing options * ⓘ	Azure load balancer
Select a load balancer * ⓘ	(new) myVMSS2021-lb <a href="#">Create new</a>
Select a backend pool * ⓘ	(new) bepool <a href="#">Create new</a>

18. On the **Create a load balancer** page, type in a name for your load balancer and **Public IP address name**.
19. For **Domain name label**, type in a name to use as a prefix for your domain name. This name must be unique.
20. When you are done, select **Create**.

## Create a load balancer X

Azure Load Balancer enables you to scale your applications and create high availability for your services. Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications. [Learn more about Azure Load Balancer. ↗](#)

Your load balancer will be placed in the same subscription, resource group, and region as your virtual machine scale set. Azure will configure basic settings for the frontend IP, backend address pools, NAT rules, and NAT pools for this load balancer automatically.

Name * ⓘ	myVMSS-lb
Public IP address name * ⓘ	myVMSS-ip
Domain name label ⓘ	myvmss2021 <span style="float: right;">✓ .eastus.cloudapp.azure.com</span>
SKU	Standard
Type	Public
Availability zone ⓘ	Zone-redundant

**Create** **Discard**

21. Back on the **Networking** tab, leave the default name for the backend pool.
22. On the **Scaling** tab, leave the default instance count as *2*, or add in your own value. This is the number of VMs that will be created, so be aware of the costs and the limits on your subscription if you change this value.
23. Leave the **Scaling policy** set to *Manual*.

An Azure virtual machine scale set can automatically increase or decrease the number of VM instances that run your application. This automated and elastic behavior reduces the management overhead to monitor and optimize the performance of your application. [Learn more about VMSS scaling](#)

Initial instance count \* ⓘ

2

#### Scaling

Scaling policy ⓘ

Manual

Custom

#### Scale-In policy

Configure the order in which virtual machines are selected for deletion during a scale-in operation.

[Learn more about scale-in policies](#)

Scale-in policy

Default - Balance across availability zones and fault domains, then delete ...



Scale-in policy is not supported for virtual machine scale sets with flexible orchestration mode (preview).

24. Select the **Advanced** tab.

25. Under **Custom data and cloud init**, copy the following and paste it into the **Custom data** text box:

```

#cloud-config
package_upgrade: true
packages:
- nginx
- nodejs
- npm
write_files:
- owner: www-data:www-data
- path: /etc/nginx/sites-available/default
  content: |
    server {
      listen 80;
      location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection keep-alive;
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
      }
    }
- owner: azureuser:azureuser
- path: /home/azureuser/myapp/index.js
  content: |
    var express = require('express')
    var app = express()
    var os = require('os');
    app.get('/', function (req, res) {
      res.send('Hello World from host ' + os.hostname() + '!')
    })
    app.listen(3000, function () {
      console.log('Hello world app listening on port 3000!')
    })
runcmd:
- service nginx restart
- cd "/home/azureuser/myapp"
- npm init
- npm install express -y
- nodejs index.js

```

26. When you are done, select **Review + create**.
27. Once you see that validation has passed, you can select **Create** at the bottom of the page to deploy your scale set.
28. When the **Generate new key pair** window opens, select **Download private key and create resource**. Your key file will be download as **myKey.pem**. Make sure you know where the **.pem** file was downloaded, you will need the path to it in the next step.
29. When the deployment is complete, select **Go to resource** to see your scale set.

## View the VMs in your scale set

On the page for the scale set, select **Instances** from the left menu.

You will see a list of VMs that are part of your scale set. This list includes:

- The name of the VM
- The computer name used by the VM.
- The current status of the VM, like *Running*.
- The *Provisioning state* of the VM, like *Succeeded*.

Name	Computer name	Status	Provisioning state
myVMSS_351a2d68	myvmss6nEQF9OO	Running	Succeeded
myVMSS_c4dac4c9	myvmss6nGESCAF	Running	Succeeded

## Open port 80

Open port 80 on your scale set by adding an inbound rule to your network security group (NSG).

1. On the page for your scale set, select **Networking** from the left menu. The **Networking** page will open.
2. Select **Add inbound port rule**. The **Add inbound security rule** page will open.
3. Under **Service**, select *HTTP* and then select **Add** at the bottom of the page.

## Test your scale set

Test your scale set by connecting to it from a browser.

1. On the **Overview** page for your scale set, copy the Public IP address.
2. Open another tab in your browser and paste the IP address into the address bar.
3. When the page loads, take a note of the compute name that is shown.
4. Refresh the page until you see the computer name change.

## Delete your scale set

When you are done, you should delete the resource group, which will delete everything you deployed for your scale set.

1. On the page for your scale set, select the **Resource group**. The page for your resource group will open.
2. At the top of the page, select **Delete resource group**.
3. In the **Are you sure you want to delete** page, type in the name of your resource group and then select **Delete**.

## Next steps

In this tutorial, you created a virtual machine scale set. You learned how to:

- Create a resource group.
- Create a Flexible scale set with a load balancer.
- Add nginx to the scale set instances.
- Open port 80 to HTTP traffic.
- Test the scale set.

Advance to the next tutorial to learn more about load balancing concepts for virtual machines.

[Load balance virtual machines](#)

# Tutorial: Load balance VMs for high availability

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Load balancing provides a higher level of availability by spreading incoming requests across multiple virtual machines. In this tutorial, you learn about the different components of the Azure load balancer that distribute traffic and provide high availability. You learn how to:

- Create a load balancer
- Create a health probe
- Create traffic rules
- Use cloud-init to install a basic Node.js app
- Create virtual machines and attach them to the load balancer
- View the load balancer in action
- Add and remove VMs from the load balancer

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Azure load balancer overview

An Azure load balancer is a Layer-4 (TCP, UDP) load balancer that provides high availability by distributing incoming traffic among healthy VMs. A load balancer health probe monitors a given port on each VM and only distributes traffic to an operational VM.

You define a front-end IP configuration that contains one or more public IP addresses. This front-end IP configuration allows your load balancer and applications to be accessible over the Internet.

Virtual machines connect to a load balancer using their virtual network interface card (NIC). To distribute traffic to the VMs, a back-end address pool contains the IP addresses of the virtual (NICs) connected to the load balancer.

To control the flow of traffic, you define load balancer rules for specific ports and protocols that map to your VMs.

If you followed the previous tutorial to [create a virtual machine scale set](#), a load balancer was created for you. All these components were configured for you as part of the scale set.

## Create Azure load balancer

This section details how you can create and configure each component of the load balancer. Before you can create your load balancer, create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroupLoadBalancer* in the *eastus* location:

```
az group create --name myResourceGroupLoadBalancer --location eastus
```

### Create a public IP address

To access your app on the Internet, you need a public IP address for the load balancer. Create a public IP address with [az network public-ip create](#). The following example creates a public IP address named *myPublicIP* in the *myResourceGroupLoadBalancer* resource group:

```
az network public-ip create \
--resource-group myResourceGroupLoadBalancer \
--name myPublicIP
```

## Create a load balancer

Create a load balancer with [az network lb create](#). The following example creates a load balancer named *myLoadBalancer* and assigns the *myPublicIP* address to the front-end IP configuration:

```
az network lb create \
--resource-group myResourceGroupLoadBalancer \
--name myLoadBalancer \
--frontend-ip-name myFrontEndPool \
--backend-pool-name myBackEndPool \
--public-ip-address myPublicIP
```

## Create a health probe

To allow the load balancer to monitor the status of your app, you use a health probe. The health probe dynamically adds or removes VMs from the load balancer rotation based on their response to health checks. By default, a VM is removed from the load balancer distribution after two consecutive failures at 15-second intervals. You create a health probe based on a protocol or a specific health check page for your app.

The following example creates a TCP probe. You can also create custom HTTP probes for more fine grained health checks. When using a custom HTTP probe, you must create the health check page, such as *healthcheck.js*. The probe must return an HTTP 200 OK response for the load balancer to keep the host in rotation.

To create a TCP health probe, you use [az network lb probe create](#). The following example creates a health probe named *myHealthProbe*:

```
az network lb probe create \
--resource-group myResourceGroupLoadBalancer \
--lb-name myLoadBalancer \
--name myHealthProbe \
--protocol tcp \
--port 80
```

## Create a load balancer rule

A load balancer rule is used to define how traffic is distributed to the VMs. You define the front-end IP configuration for the incoming traffic and the back-end IP pool to receive the traffic, along with the required source and destination port. To make sure only healthy VMs receive traffic, you also define the health probe to use.

Create a load balancer rule with [az network lb rule create](#). The following example creates a rule named *myLoadBalancerRule*, uses the *myHealthProbe* health probe, and balances traffic on port 80:

```
az network lb rule create \
--resource-group myResourceGroupLoadBalancer \
--lb-name myLoadBalancer \
--name myLoadBalancerRule \
--protocol tcp \
--frontend-port 80 \
--backend-port 80 \
--frontend-ip-name myFrontEndPool \
--backend-pool-name myBackEndPool \
--probe-name myHealthProbe
```

## Configure virtual network

Before you deploy some VMs and can test your balancer, create the supporting virtual network resources. For more information about virtual networks, see the [Manage Azure Virtual Networks](#) tutorial.

### Create network resources

Create a virtual network with [az network vnet create](#). The following example creates a virtual network named *myVnet* with a subnet named *mySubnet*.

```
az network vnet create \
--resource-group myResourceGroupLoadBalancer \
--name myVnet \
--subnet-name mySubnet
```

To add a network security group, you use [az network nsg create](#). The following example creates a network security group named *myNetworkSecurityGroup*.

```
az network nsg create \
--resource-group myResourceGroupLoadBalancer \
--name myNetworkSecurityGroup
```

Create a network security group rule with [az network nsg rule create](#). The following example creates a network security group rule named *myNetworkSecurityGroupRule*.

```
az network nsg rule create \
--resource-group myResourceGroupLoadBalancer \
--nsg-name myNetworkSecurityGroup \
--name myNetworkSecurityGroupRule \
--priority 1001 \
--protocol tcp \
--destination-port-range 80
```

Virtual NICs are created with [az network nic create](#). The following example creates three virtual NICs. (One virtual NIC for each VM you create for your app in the following steps). You can create additional virtual NICs and VMs at any time and add them to the load balancer:

```
for i in `seq 1 3`; do
    az network nic create \
        --resource-group myResourceGroupLoadBalancer \
        --name myNic$i \
        --vnet-name myVnet \
        --subnet mySubnet \
        --network-security-group myNetworkSecurityGroup \
        --lb-name myLoadBalancer \
        --lb-address-pools myBackEndPool
done
```

When all three virtual NICs are created, continue on to the next step

## Create virtual machines

### Create cloud-init config

In a previous tutorial on [How to customize a Linux virtual machine on first boot](#), you learned how to automate VM customization with cloud-init. You can use the same cloud-init configuration file to install NGINX and run a simple 'Hello World' Node.js app in the next step. To see the load balancer in action, at the end of the tutorial you access this simple app in a web browser.

In your current shell, create a file named *cloud-init.txt* and paste the following configuration. For example, create the file in the Cloud Shell not on your local machine. Enter `sensible-editor cloud-init.txt` to create the file and see a list of available editors. Make sure that the whole cloud-init file is copied correctly, especially the first line:

```

#cloud-config
package_upgrade: true
packages:
- nginx
- nodejs
- npm
write_files:
- owner: www-data:www-data
- path: /etc/nginx/sites-available/default
  content: |
    server {
      listen 80;
      location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection keep-alive;
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
      }
    }
- owner: azureuser:azureuser
- path: /home/azureuser/myapp/index.js
  content: |
    var express = require('express')
    var app = express()
    var os = require('os');
    app.get('/', function (req, res) {
      res.send('Hello World from host ' + os.hostname() + '!')
    })
    app.listen(3000, function () {
      console.log('Hello world app listening on port 3000!')
    })
runcmd:
- service nginx restart
- cd "/home/azureuser/myapp"
- npm init
- npm install express -y
- nodejs index.js

```

## Create virtual machines

To improve the high availability of your app, place your VMs in an availability set. For more information about availability sets, see the previous [How to create highly available virtual machines](#) tutorial.

Create an availability set with [az vm availability-set create](#). The following example creates an availability set named *myAvailabilitySet*.

```

az vm availability-set create \
--resource-group myResourceGroupLoadBalancer \
--name myAvailabilitySet

```

Now you can create the VMs with [az vm create](#). The following example creates three VMs and generates SSH keys if they do not already exist:

```

for i in `seq 1 3`; do
    az vm create \
        --resource-group myResourceGroupLoadBalancer \
        --name myVM$i \
        --availability-set myAvailabilitySet \
        --nics myNic$i \
        --image UbuntuLTS \
        --admin-username azureuser \
        --generate-ssh-keys \
        --custom-data cloud-init.txt \
        --no-wait
done

```

There are background tasks that continue to run after the Azure CLI returns you to the prompt. The `--no-wait` parameter does not wait for all the tasks to complete. It may be another couple of minutes before you can access the app. The load balancer health probe automatically detects when the app is running on each VM. Once the app is running, the load balancer rule starts to distribute traffic.

## Test load balancer

Obtain the public IP address of your load balancer with [az network public-ip show](#). The following example obtains the IP address for *myPublicIP* created earlier:

```

az network public-ip show \
    --resource-group myResourceGroupLoadBalancer \
    --name myPublicIP \
    --query [ipAddress] \
    --output tsv

```

You can then enter the public IP address in to a web browser. Remember - it takes a few minutes for the VMs to be ready before the load balancer starts to distribute traffic to them. The app is displayed, including the hostname of the VM that the load balancer distributed traffic to as in the following example:



To see the load balancer distribute traffic across all three VMs running your app, you can force-refresh your web browser.

## Add and remove VMs

You may need to perform maintenance on the VMs running your app, such as installing OS updates. To deal with increased traffic to your app, you may need to add additional VMs. This section shows you how to remove or add a VM from the load balancer.

### Remove a VM from the load balancer

You can remove a VM from the backend address pool with [az network nic ip-config address-pool remove](#). The following example removes the virtual NIC for *myVM2* from *myLoadBalancer*.

```
az network nic ip-config address-pool remove \
--resource-group myResourceGroupLoadBalancer \
--nic-name myNic2 \
--ip-config-name ipConfig1 \
--lb-name myLoadBalancer \
--address-pool myBackEndPool
```

To see the load balancer distribute traffic across the remaining two VMs running your app you can force-refresh your web browser. You can now perform maintenance on the VM, such as installing OS updates or performing a VM reboot.

To view a list of VMs with virtual NICs connected to the load balancer, use [az network lb address-pool show](#). Query and filter on the ID of the virtual NIC as follows:

```
az network lb address-pool show \
--resource-group myResourceGroupLoadBalancer \
--lb-name myLoadBalancer \
--name myBackEndPool \
--query backendIpConfigurations \
--output tsv | cut -f4
```

The output is similar to the following example, which shows that the virtual NIC for VM 2 is no longer part of the backend address pool:

```
/subscriptions/<guid>/resourceGroups/myResourceGroupLoadBalancer/providers/Microsoft.Network/networkInterfaces/myNic1/ipConfigurations/ipconfig1
/subscriptions/<guid>/resourceGroups/myResourceGroupLoadBalancer/providers/Microsoft.Network/networkInterfaces/myNic3/ipConfigurations/ipconfig1
```

## Add a VM to the load balancer

After performing VM maintenance, or if you need to expand capacity, you can add a VM to the backend address pool with [az network nic ip-config address-pool add](#). The following example adds the virtual NIC for *myVM2* to *myLoadBalancer*.

```
az network nic ip-config address-pool add \
--resource-group myResourceGroupLoadBalancer \
--nic-name myNic2 \
--ip-config-name ipConfig1 \
--lb-name myLoadBalancer \
--address-pool myBackEndPool
```

To verify that the virtual NIC is connected to the backend address pool, use [az network lb address-pool show](#) again from the preceding step.

## Next steps

In this tutorial, you created a load balancer and attached VMs to it. You learned how to:

- Create an Azure load balancer
- Create a load balancer health probe
- Create load balancer traffic rules
- Use cloud-init to create a basic Node.js app
- Create virtual machines and attach to a load balancer
- View a load balancer in action

- Add and remove VMs from a load balancer

Advance to the next tutorial to learn more about Azure virtual network components.

[Manage VMs and virtual networks](#)

# Tutorial: Create and manage Azure virtual networks for Linux virtual machines with the Azure CLI

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure virtual machines use Azure networking for internal and external network communication. This tutorial walks through deploying two virtual machines and configuring Azure networking for these VMs. The examples in this tutorial assume that the VMs are hosting a web application with a database back-end, however an application is not deployed in the tutorial. In this tutorial, you learn how to:

- Create a virtual network and subnet
- Create a public IP address
- Create a front-end VM
- Secure network traffic
- Create a back-end VM

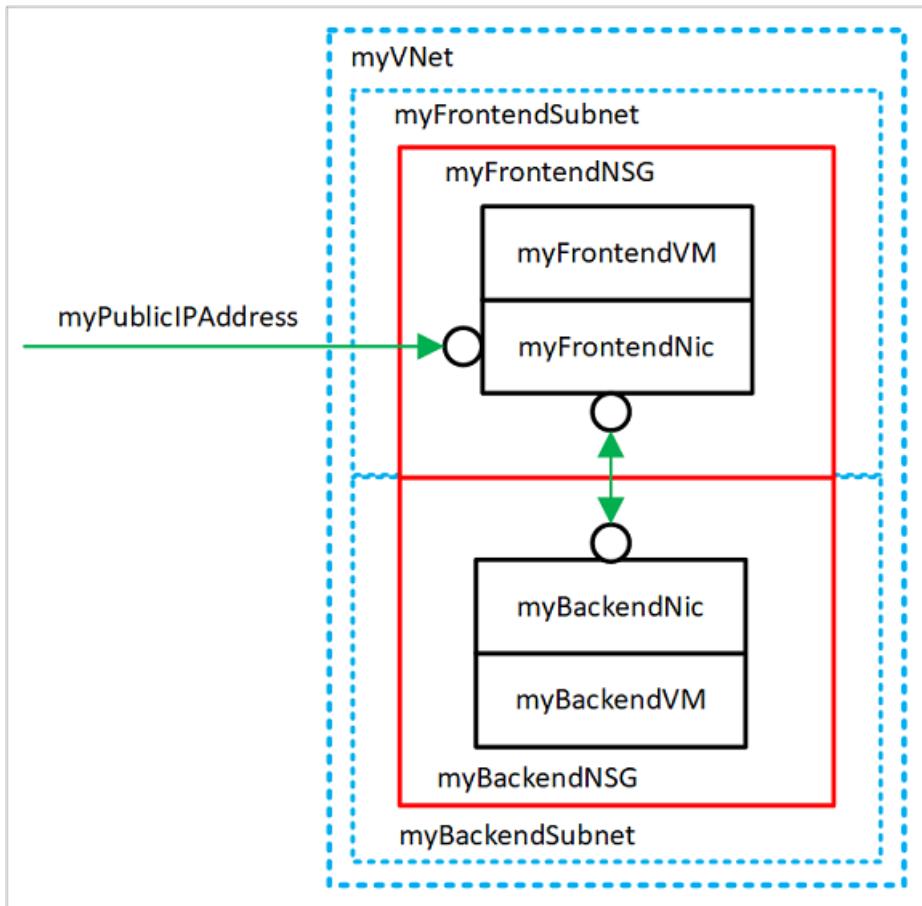
This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## VM networking overview

Azure virtual networks enable secure network connections between virtual machines, the internet, and other Azure services such as Azure SQL Database. Virtual networks are broken down into logical segments called subnets. Subnets are used to control network flow, and as a security boundary. When deploying a VM, it generally includes a virtual network interface, which is attached to a subnet.

As you complete the tutorial, the following virtual network resources are created:



- *myVNet* - The virtual network that the VMs use to communicate with each other and the internet.
- *myFrontendSubnet* - The subnet in *myVNet* used by the front-end resources.
- *myPublicIPAddress* - The public IP address used to access *myFrontendVM* from the internet.
- *myFrontendNic* - The network interface used by *myFrontendVM* to communicate with *myBackendVM*.
- *myFrontendVM* - The VM used to communicate between the internet and *myBackendVM*.
- *myBackendNSG* - The network security group that controls communication between the *myFrontendVM* and *myBackendVM*.
- *myBackendSubnet* - The subnet associated with *myBackendNSG* and used by the back-end resources.
- *myBackendNic* - The network interface used by *myBackendVM* to communicate with *myFrontendVM*.
- *myBackendVM* - The VM that uses port 22 and 3306 to communicate with *myFrontendVM*.

## Create a virtual network and subnet

For this tutorial, a single virtual network is created with two subnets. A front-end subnet for hosting a web application, and a back-end subnet for hosting a database server.

Before you can create a virtual network, create a resource group with [az group create](#). The following example creates a resource group named *myRGNetwork* in the *eastus* location.

```
az group create --name myRGNetwork --location eastus
```

### Create virtual network

Use the [az network vnet create](#) command to create a virtual network. In this example, the network is named *mvVNet* and is given an address prefix of *10.0.0.0/16*. A subnet is also created with a name of *myFrontendSubnet* and a prefix of *10.0.1.0/24*. Later in this tutorial a front-end VM is connected to this subnet.

```
az network vnet create \
--resource-group myRGNetwork \
--name myVNet \
--address-prefix 10.0.0.0/16 \
--subnet-name myFrontendSubnet \
--subnet-prefix 10.0.1.0/24
```

## Create subnet

A new subnet is added to the virtual network using the [az network vnet subnet create](#) command. In this example, the subnet is named *myBackendSubnet* and is given an address prefix of *10.0.2.0/24*. This subnet is used with all back-end services.

```
az network vnet subnet create \
--resource-group myRGNetwork \
--vnet-name myVNet \
--name myBackendSubnet \
--address-prefix 10.0.2.0/24
```

At this point, a network has been created and segmented into two subnets, one for front-end services, and another for back-end services. In the next section, virtual machines are created and connected to these subnets.

## Create a public IP address

A public IP address allows Azure resources to be accessible on the internet. The allocation method of the public IP address can be configured as dynamic or static. By default, a public IP address is dynamically allocated. Dynamic IP addresses are released when a VM is deallocated. This behavior causes the IP address to change during any operation that includes a VM deallocation.

The allocation method can be set to static, which ensures that the IP address remains assigned to a VM, even during a deallocated state. When using a statically allocated IP address, the IP address itself cannot be specified. Instead, it is allocated from a pool of available addresses.

```
az network public-ip create --resource-group myRGNetwork --name myPublicIPAddress
```

When creating a VM with the [az vm create](#) command, the default public IP address allocation method is dynamic. When creating a virtual machine using the [az vm create](#) command, include the `--public-ip-address-allocation static` argument to assign a static public IP address. This operation is not demonstrated in this tutorial, however in the next section a dynamically allocated IP address is changed to a statically allocated address.

## Change allocation method

The IP address allocation method can be changed using the [az network public-ip update](#) command. In this example, the IP address allocation method of the front-end VM is changed to static.

First, deallocate the VM.

```
az vm deallocate --resource-group myRGNetwork --name myFrontendVM
```

Use the [az network public-ip update](#) command to update the allocation method. In this case, the `--allocation-method` is being set to *static*.

```
az network public-ip update --resource-group myRGNetwork --name myPublicIPAddress --allocation-method static
```

Start the VM.

```
az vm start --resource-group myRGNetwork --name myFrontendVM --no-wait
```

## No public IP address

Often, a VM does not need to be accessible over the internet. To create a VM without a public IP address, use the `--public-ip-address ""` argument with an empty set of double quotes. This configuration is demonstrated later in this tutorial.

## Create a front-end VM

Use the `az vm create` command to create the VM named *myFrontendVM* using *myPublicIPAddress*.

```
az vm create \
--resource-group myRGNetwork \
--name myFrontendVM \
--vnet-name myVNet \
--subnet myFrontendSubnet \
--nsg myFrontendNSG \
--public-ip-address myPublicIPAddress \
--image UbuntuLTS \
--generate-ssh-keys
```

## Secure network traffic

A network security group (NSG) contains a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). NSGs can be associated to subnets or individual network interfaces. When an NSG is associated with a network interface, it applies only to the associated VM. When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet.

### Network security group rules

NSG rules define networking ports over which traffic is allowed or denied. The rules can include source and destination IP address ranges so that traffic is controlled between specific systems or subnets. NSG rules also include a priority (between 1—and 4096). Rules are evaluated in the order of priority. A rule with a priority of 100 is evaluated before a rule with priority 200.

All NSGs contain a set of default rules. The default rules cannot be deleted, but because they are assigned the lowest priority, they can be overridden by the rules that you create.

The default rules for NSGs are:

- **Virtual network** - Traffic originating and ending in a virtual network is allowed both in inbound and outbound directions.
- **Internet** - Outbound traffic is allowed, but inbound traffic is blocked.
- **Load balancer** - Allow Azure's load balancer to probe the health of your VMs and role instances. If you are not using a load balanced set, you can override this rule.

### Create network security groups

A network security group can be created at the same time as a VM using the `az vm create` command. When doing so, the NSG is associated with the VMs network interface and an NSG rule is auto created to allow traffic on port 22 from any source. Earlier in this tutorial, the front-end NSG was auto-created with the front-end VM. An NSG rule was also auto created for port 22.

In some cases, it may be helpful to pre-create an NSG, such as when default SSH rules should not be created, or when the NSG should be attached to a subnet.

Use the [az network nsg create](#) command to create a network security group.

```
az network nsg create --resource-group myRGNetwork --name myBackendNSG
```

Instead of associating the NSG to a network interface, it is associated with a subnet. In this configuration, any VM that is attached to the subnet inherits the NSG rules.

Update the existing subnet named *myBackendSubnet* with the new NSG.

```
az network vnet subnet update \
--resource-group myRGNetwork \
--vnet-name myVNet \
--name myBackendSubnet \
--network-security-group myBackendNSG
```

## Secure incoming traffic

When the front-end VM was created, an NSG rule was created to allow incoming traffic on port 22. This rule allows SSH connections to the VM. For this example, traffic should also be allowed on port *80*. This configuration allows a web application to be accessed on the VM.

Use the [az network nsg rule create](#) command to create a rule for port *80*.

```
az network nsg rule create \
--resource-group myRGNetwork \
--nsg-name myFrontendNSG \
--name http \
--access allow \
--protocol Tcp \
--direction Inbound \
--priority 200 \
--source-address-prefix "*" \
--source-port-range "*" \
--destination-address-prefix "*" \
--destination-port-range 80
```

The front-end VM is only accessible on port 22 and port *80*. All other incoming traffic is blocked at the network security group. It may be helpful to visualize the NSG rule configurations. Return the NSG rule configuration with the [az network rule list](#) command.

```
az network nsg rule list --resource-group myRGNetwork --nsg-name myFrontendNSG --output table
```

## Secure VM to VM traffic

Network security group rules can also apply between VMs. For this example, the front-end VM needs to communicate with the back-end VM on port 22 and *3306*. This configuration allows SSH connections from the front-end VM, and also allow an application on the front-end VM to communicate with a back-end MySQL database. All other traffic should be blocked between the front-end and back-end virtual machines.

Use the [az network nsg rule create](#) command to create a rule for port 22. Notice that the

--source-address-prefix argument specifies a value of *10.0.1.0/24*. This configuration ensures that only traffic from the front-end subnet is allowed through the NSG.

```
az network nsg rule create \
--resource-group myRGNetwork \
--nsg-name myBackendNSG \
--name SSH \
--access Allow \
--protocol Tcp \
--direction Inbound \
--priority 100 \
--source-address-prefix 10.0.1.0/24 \
--source-port-range "*" \
--destination-address-prefix "*" \
--destination-port-range "22"
```

Now add a rule for MySQL traffic on port 3306.

```
az network nsg rule create \
--resource-group myRGNetwork \
--nsg-name myBackendNSG \
--name MySQL \
--access Allow \
--protocol Tcp \
--direction Inbound \
--priority 200 \
--source-address-prefix 10.0.1.0/24 \
--source-port-range "*" \
--destination-address-prefix "*" \
--destination-port-range "3306"
```

Finally, because NSGs have a default rule allowing all traffic between VMs in the same VNet, a rule can be created for the back-end NSGs to block all traffic. Notice here that the `--priority` is given a value of *300*, which is lower than both the NSG and MySQL rules. This configuration ensures that SSH and MySQL traffic is still allowed through the NSG.

```
az network nsg rule create \
--resource-group myRGNetwork \
--nsg-name myBackendNSG \
--name denyAll \
--access Deny \
--protocol Tcp \
--direction Inbound \
--priority 300 \
--source-address-prefix "*" \
--source-port-range "*" \
--destination-address-prefix "*" \
--destination-port-range "*"
```

## Create back-end VM

Now create a virtual machine, which is attached to the *myBackendSubnet*. Notice that the `--nsg` argument has a value of empty double quotes. An NSG does not need to be created with the VM. The VM is attached to the back-end subnet, which is protected with the pre-created back-end NSG. This NSG applies to the VM. Also, notice here that the `--public-ip-address` argument has a value of empty double quotes. This configuration creates a VM without a public IP address.

```
az vm create \
--resource-group myRGNetwork \
--name myBackendVM \
--vnet-name myVNet \
--subnet myBackendSubnet \
--public-ip-address "" \
--nsg "" \
--image UbuntuLTS \
--generate-ssh-keys
```

The back-end VM is only accessible on port 22 and port 3306 from the front-end subnet. All other incoming traffic is blocked at the network security group. It may be helpful to visualize the NSG rule configurations. Return the NSG rule configuration with the [az network rule list](#) command.

```
az network nsg rule list --resource-group myRGNetwork --nsg-name myBackendNSG --output table
```

## Next steps

In this tutorial, you created and secured Azure networks as related to virtual machines. You learned how to:

- Create a virtual network and subnet
- Create a public IP address
- Create a front-end VM
- Secure network traffic
- Create back-end VM

To learn about protecting your VM disks, see [Backup and disaster recovery for disks](#).

# Tutorial: Create and Manage Windows VMs with Azure PowerShell

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Azure virtual machines provide a fully configurable and flexible computing environment. This tutorial covers basic Azure virtual machine (VM) deployment tasks like selecting a VM size, selecting a VM image, and deploying a VM. You learn how to:

- Create and connect to a VM
- Select and use VM images
- View and use specific VM sizes
- Resize a VM
- View and understand VM state

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create resource group

Create a resource group with the [New-AzResourceGroup](#) command.

An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine. In the following example, a resource group named *myResourceGroupVM* is created in the *EastUS* region:

```
New-AzResourceGroup  
  -ResourceGroupName "myResourceGroupVM"  
  -Location "EastUS"
```

The resource group is specified when creating or modifying a VM, which can be seen throughout this tutorial.

## Create a VM

When creating a VM, several options are available like operating system image, network configuration, and administrative credentials. This example creates a VM named *myVM*, running the default version of Windows Server 2016 Datacenter.

Set the username and password needed for the administrator account on the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Create the VM with [New-AzVM](#).

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroupVM" ` 
    -Name "myVM" ` 
    -Location "EastUS" ` 
    -VirtualNetworkName "myVnet" ` 
    -SubnetName "mySubnet" ` 
    -SecurityGroupName "myNetworkSecurityGroup" ` 
    -PublicIpAddressName "myPublicIpAddress" ` 
    -Credential $cred
```

## Connect to VM

After the deployment has completed, create a remote desktop connection with the VM.

Run the following commands to return the public IP address of the VM. Take note of this IP Address so you can connect to it with your browser to test web connectivity in a future step.

```
Get-AzPublicIpAddress ` 
    -ResourceGroupName "myResourceGroupVM" | Select IpAddress
```

Use the following command, on your local machine, to create a remote desktop session with the VM. Replace the IP address with the *publicIpAddress* of your VM. When prompted, enter the credentials used when creating the VM.

```
mstsc /v:<publicIpAddress>
```

In the **Windows Security** window, select **More choices** and then **Use a different account**. Type the username and password you created for the VM and then click **OK**.

## Understand marketplace images

The Azure marketplace includes many images that can be used to create a new VM. In the previous steps, a VM was created using the Windows Server 2016 Datacenter image. In this step, the PowerShell module is used to search the marketplace for other Windows images, which can also be used as a base for new VMs. This process consists of finding the publisher, offer, SKU, and optionally a version number to **identify** the image.

Use the [Get-AzVMImagePublisher](#) command to return a list of image publishers:

```
Get-AzVMImagePublisher -Location "EastUS"
```

Use the [Get-AzVMImageOffer](#) to return a list of image offers. With this command, the returned list is filtered on the specified publisher named `MicrosoftWindowsServer`:

```
Get-AzVMImageOffer ` 
    -Location "EastUS" ` 
    -PublisherName "MicrosoftWindowsServer"
```

The results will look something like this example:

Offer	PublisherName	Location
Windows-HUB	MicrosoftWindowsServer	EastUS
WindowsServer	MicrosoftWindowsServer	EastUS
WindowsServer-HUB	MicrosoftWindowsServer	EastUS

The [Get-AzVMImageSku](#) command will then filter on the publisher and offer name to return a list of image names.

```
Get-AzVMImageSku ` 
    -Location "EastUS" ` 
    -PublisherName "MicrosoftWindowsServer" ` 
    -Offer "WindowsServer"
```

The results will look something like this example:

Skus	Offer	PublisherName	Location
2008-R2-SP1	WindowsServer	MicrosoftWindowsServer	EastUS
2008-R2-SP1-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2012-Datacenter	WindowsServer	MicrosoftWindowsServer	EastUS
2012-Datacenter-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2012-R2-Datacenter	WindowsServer	MicrosoftWindowsServer	EastUS
2012-R2-Datacenter-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-Server-Core	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-Server-Core-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-with-Containers	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-with-Containers-smalldisk	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Datacenter-with-RDSH	WindowsServer	MicrosoftWindowsServer	EastUS
2016-Nano-Server	WindowsServer	MicrosoftWindowsServer	EastUS

This information can be used to deploy a VM with a specific image. This example deploys a VM using the latest version of a Windows Server 2016 with Containers image.

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroupVM" ` 
    -Name "myVM2" ` 
    -Location "EastUS" ` 
    -VirtualNetworkName "myVnet" ` 
    -SubnetName "mySubnet" ` 
    -SecurityGroupName "myNetworkSecurityGroup" ` 
    -PublicIpAddressName "myPublicIpAddress2" ` 
    -ImageName "MicrosoftWindowsServer:WindowsServer:2016-Datacenter-with-Containers:latest" ` 
    -Credential $cred ` 
    -AsJob
```

The `-AsJob` parameter creates the VM as a background task, so the PowerShell prompts return to you. You can view details of background jobs with the [Get-Job](#) cmdlet.

## Understand VM sizes

The VM size determines the amount of compute resources like CPU, GPU, and memory that are made available to the VM. Virtual machines should be created using a VM size appropriate for the workload. If a workload increases, an existing virtual machine can also be resized.

### VM Sizes

The following table categorizes sizes into use cases.

Type	Common Sizes	Description
General purpose	B, Dsv3, Dv3, DSv2, Dv2, Av2, DC	Balanced CPU-to-memory. Ideal for dev / test and small to medium applications and data solutions.
Compute optimized	Fsv2	High CPU-to-memory. Good for medium traffic applications, network appliances, and batch processes.
Memory optimized	Esv3, Ev3, M, DSv2, Dv2	High memory-to-core. Great for relational databases, medium to large caches, and in-memory analytics.
Storage optimized	Lsv2, Ls	High disk throughput and IO. Ideal for Big Data, SQL, and NoSQL databases.
GPU	NV, NVv2, NC, NCv2, NCv3, ND	Specialized VMs targeted for heavy graphic rendering and video editing.
High performance	H	Our most powerful CPU VMs with optional high-throughput network interfaces (RDMA).

## Find available VM sizes

To see a list of VM sizes available in a particular region, use the [Get-AzVMSize](#) command.

```
Get-AzVMSize -Location "EastUS"
```

## Resize a VM

After a VM has been deployed, it can be resized to increase or decrease resource allocation.

Before resizing a VM, check if the size you want is available on the current VM cluster. The [Get-AzVMSize](#) command returns a list of sizes.

```
Get-AzVMSize -ResourceGroupName "myResourceGroupVM" -VMName "myVM"
```

If the size is available, the VM can be resized from a powered-on state, however it is rebooted during the operation.

```
$vm = Get-AzVM ` 
-ResourceGroupName "myResourceGroupVM" ` 
-VMName "myVM"
$vm.HardwareProfile.VmSize = "Standard_DS3_v2"
Update-AzVM ` 
-VM $vm ` 
-ResourceGroupName "myResourceGroupVM"
```

If the size you want isn't available on the current cluster, the VM needs to be deallocated before the resize operation can occur. Deallocation of a VM will remove any data on the temp disk, and the public IP address will change unless a static IP address is being used.

```

Stop-AzVM ` 
  -ResourceGroupName "myResourceGroupVM" ` 
  -Name "myVM" -Force
$vm = Get-AzVM ` 
  -ResourceGroupName "myResourceGroupVM" ` 
  -VMName "myVM"
$vm.HardwareProfile.VmSize = "Standard_E2s_v3"
Update-AzVM -VM $vm ` 
  -ResourceGroupName "myResourceGroupVM"
Start-AzVM ` 
  -ResourceGroupName "myResourceGroupVM" ` 
  -Name $vm.name

```

## VM power states

An Azure VM can have one of many power states.

POWER STATE	DESCRIPTION
Starting	The virtual machine is being started.
Running	The virtual machine is running.
Stopping	The virtual machine is being stopped.
Stopped	The VM is stopped. Virtual machines in the stopped state still incur compute charges.
Deallocating	The VM is being deallocated.
Deallocated	Indicates that the VM is removed from the hypervisor but is still available in the control plane. Virtual machines in the <code>Deallocated</code> state do not incur compute charges.
-	The power state of the VM is unknown.

To get the state of a particular VM, use the [Get-AzVM](#) command. Be sure to specify a valid name for a VM and resource group.

```

Get-AzVM ` 
  -ResourceGroupName "myResourceGroupVM" ` 
  -Name "myVM" ` 
  -Status | Select @{n="Status"; e={$_.Statuses[1].Code}}

```

The output will look something like this example:

```

Status
-----
PowerState/running

```

To retrieve the power state of all the VMs in your subscription, use the [Virtual Machines - List All API](#) with parameter `statusOnly` set to `true`.

## Management tasks

During the lifecycle of a VM, you may want to run management tasks like starting, stopping, or deleting a VM. Additionally, you may want to create scripts to automate repetitive or complex tasks. Using Azure PowerShell, many common management tasks can be run from the command line or in scripts.

## Stop a VM

Stop and deallocate a VM with [Stop-AzVM](#):

```
Stop-AzVM ` 
-ResourceGroupName "myResourceGroupVM" ` 
-Name "myVM" -Force
```

If you want to keep the VM in a provisioned state, use the `-StayProvisioned` parameter.

## Start a VM

```
Start-AzVM ` 
-ResourceGroupName "myResourceGroupVM" ` 
-Name "myVM"
```

## Deleting VM resources

You can delete a VM, but by default this only deletes the VM resource, not the disks and networking resources the VM uses. You can change the default behavior to delete other resources when you delete the VM. For more information, see [Delete a VM and attached resources](#).

## Next steps

In this tutorial, you learned about basic VM creation and management such as how to:

- Create and connect to a VM
- Select and use VM images
- View and use specific VM sizes
- Resize a VM
- View and understand VM state

Advance to the next tutorial to learn about VM disks.

[Create and Manage VM disks](#)

# Tutorial: Manage disks with Azure PowerShell

9/21/2022 • 12 minutes to read • [Edit Online](#)

Azure virtual machines (VMs) use disks to store operating systems (OS), applications, and data. When you create a VM, it's important to choose an appropriate disk size and configuration for the expected workload.

This tutorial covers deployment and management of VM disks. In this tutorial, you learn how to:

- Create, attach, and initialize a data disk
- Verify a disk's status
- Initialize a disk
- Expand and upgrade a disk
- Detach and delete a disk

## Prerequisites

You must have an Azure account with an active subscription. If you don't have an Azure subscription, create a [free account](#) before you begin.

## Azure Cloud Shell

Azure hosts Azure Cloud Shell, an interactive shell environment that you can use through your browser. You can use either Bash or PowerShell with Cloud Shell to work with Azure services. You can use the Cloud Shell preinstalled commands to run the code in this article, without having to install anything on your local environment.

To start Azure Cloud Shell:

OPTION	EXAMPLE/LINK
Select <b>Try It</b> in the upper-right corner of a code or command block. Selecting <b>Try It</b> doesn't automatically copy the code or command to Cloud Shell.	
Go to <a href="https://shell.azure.com">https://shell.azure.com</a> , or select the <b>Launch Cloud Shell</b> button to open Cloud Shell in your browser.	
Select the <b>Cloud Shell</b> button on the menu bar at the upper right in the <a href="#">Azure portal</a> .	

To use Azure Cloud Shell:

1. Start Cloud Shell.
2. Select the **Copy** button on a code block (or command block) to copy the code or command.
3. Paste the code or command into the Cloud Shell session by selecting **Ctrl+Shift+V** on Windows and Linux, or by selecting **Cmd+Shift+V** on macOS.
4. Select **Enter** to run the code or command.

# Create a VM

The exercises in this tutorial require a VM. Follow the steps in this section to create one.

Before you begin, find the `$azRegion` variable located in the first line of sample code and update the value to reflect your desired region. For example, to specify the Central US region, use `$azRegion = "Central US"`. Next, use the code to deploy a VM within a new resource group. You're prompted for username and password values for the VM's local administrator account.

```
$azRegion = "[Your Region]"
$azResourceGroup = "myDemoResourceGroup"
$azVMName = "myDemoVM"
$azDataDiskName = "myDemoDataDisk"

New-AzVm ` 
    -Location $azRegion ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName ` 
    -Size "Standard_D2s_v3" ` 
    -VirtualNetworkName "myDemoVnet" ` 
    -SubnetName "myDemoSubnet" ` 
    -SecurityGroupName "myDemoNetworkSecurityGroup" ` 
    -PublicIpAddressName "myDemoPublicIpAddress"
```

The output confirms the VM's successful creation.

```
ResourceGroupName      : myDemoResourceGroup
Id                   :
/subscriptions/{GUID}/resourceGroups/myDemoResourceGroup/providers/Microsoft.Compute/virtualMachines/myDemoT
estVM
VmId                : [{GUID}]
Name                 : myDemoVM
Type                 : Microsoft.Compute/virtualMachines
Location             : centralus
Tags                 : {}
HardwareProfile       : {VmSize}
NetworkProfile        : {NetworkInterfaces}
OSProfile            : {ComputerName, AdminUsername, WindowsConfiguration, AllowExtensionOperations,
RequireGuestProvisionSignal}
ProvisioningState     : Succeeded
StorageProfile        : {ImageReference, OsDisk, DataDisks}
FullyQualifiedDomainName : mydemovm-abc123.Central US.cloudapp.azure.com
```

The VM is provisioned, and two disks are automatically created and attached.

- An **operating system disk**, which hosts the virtual machine's operating system.
- A **temporary disk**, which is primarily used for operations such as temporary data processing.

## Add a data disk

We recommend that you separate application and user data from OS-related data when possible. If you need to store user or application data on your VM, you'll typically create and attach additional data disks.

Follow the steps in this section to create, attach, and initialize a data disk on the VM.

### Create the data disk

This section guides you through the creation of a data disk.

1. Before a data disk can be created, you must first create a disk object. The following code sample uses the [New-AzDiskConfig](#) cmdlet to configure a disk object.

```
$diskConfig = New-AzDiskConfig ` 
    -Location $azRegion ` 
    -CreateOption Empty ` 
    -DiskSizeGB 128 ` 
    -SkuName "Standard_LRS"
```

2. After the disk object is created, use the [New-AzDisk](#) cmdlet to provision a data disk.

```
$dataDisk = New-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $azDataDiskName ` 
    -Disk $diskConfig
```

You can use the [Get-AzDisk](#) cmdlet to verify that the disk was created.

```
Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $azDataDiskName
```

In this example, the output confirms that the disk was created. The `DiskState` and `ManagedBy` property values confirm that the disk is not yet attached.

```
ResourceGroupName      : myDemoResourceGroup
ManagedBy              :
ManagedByExtended      : {}
OsType                 :
DiskSizeGB             : 128
DiskSizeBytes          : 137438953472
ProvisioningState      : Succeeded
DiskIOPSReadWrite      : 500
DiskMBpsReadWrite      : 60
DiskState               : Unattached
Name                   : myDemoDataDisk
```

## Attach the data disk

A data disk must be attached to a VM before the VM can access it. Complete the steps in this section to create a reference for the VM, connect the disk, and update the VM's configuration.

1. Get the VM to which you'll attach the data disk. The following sample code uses the [Get-AzVM](#) cmdlet to create a reference to the VM.

```
$vm = Get-AzVM ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName
```

2. Next, attach the data disk to the VM's configuration with the [Add-AzVMDataDisk](#) cmdlet.

```
$vm = Add-AzVMDataDisk ` 
    -VM $vm ` 
    -Name $azDataDiskName ` 
    -CreateOption Attach ` 
    -ManagedDiskId $dataDisk.Id ` 
    -Lun 1
```

3. Finally, update the VM's configuration with the [Update-AzVM](#) cmdlet.

```
Update-AzVM  
    -ResourceGroupName $azResourceGroup  
    -VM $vm
```

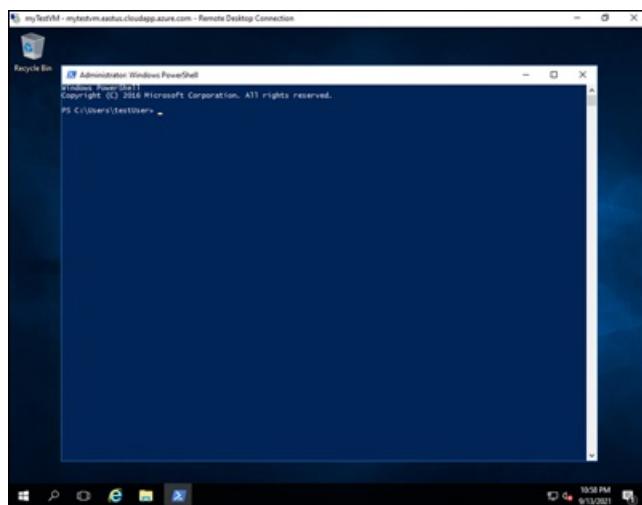
After a brief pause, the output confirms a successful attachment.

RequestId	IsSuccessStatusCode	StatusCode	ReasonPhrase
True	OK	OK	

## Initialize the data disk

After a data disk is attached to the VM, the OS needs to be configured to use the disk. The following section provides guidance on how to connect to the remote VM and configure the first disk added.

1. Sign in to the [Azure portal](#).
2. Locate the VM to which you've attached the data disk. Create a Remote Desktop Protocol (RDP) connection and sign in as the local administrator.
3. After you establish an RDP connection to the remote VM, select the Windows **Start** menu. Enter **PowerShell** in the search box and select **Windows PowerShell** to open a PowerShell window.



4. In the open PowerShell window, run the following script.

```
Get-Disk | Where PartitionStyle -eq 'raw' |  
Initialize-Disk -PartitionStyle MBR -PassThru |  
New-Partition -AssignDriveLetter -UseMaximumSize |  
Format-Volume -FileSystem NTFS -NewFileSystemLabel "myDemoDataDisk" -Confirm:$false
```

The output confirms a successful initialization.

DriveLetter	FileSystemLabel	FileSystem	DriveType	HealthStatus	OperationalStatus	SizeRemaining	Size
F	myDemoDataDisk	NTFS	Fixed	Healthy	OK	127.89 GB	128 GB

## Expand a disk

You can expand Azure disks to provide extra storage capacity when your VM is low on available disk space.

Some scenarios require data to be stored on the OS disk. For example, you may be required to support legacy

applications that install components on the OS drive. You may also have the need to migrate an on-premises physical PC or VM with a larger OS drive. In such cases, it may become necessary to expand a VM's OS disk.

Shrinking an existing disk isn't supported, and can potentially result in data loss.

## Update the disk's size

Follow the steps below to resize either the OS disk or a data disk.

1. Select the VM that contains the disk that you'll resize with the `Get-AzVM` cmdlet.

```
$vm = Get-AzVM ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName
```

2. Before you can resize a VM's disk, you must stop the VM. Use the `Stop-AzVM` cmdlet to stop the VM. You'll be prompted for confirmation.

### IMPORTANT

Before you initiate a VM shutdown, always confirm that there are no important resources or data that could be lost.

```
Stop-AzVM ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName
```

After a short pause, the output confirms that the machine is successfully stopped.

```
OperationId : abcd1234-ab12-cd34-123456abcdef
Status       : Succeeded
StartTime    : 9/13/2021 7:10:23 PM
EndTime      : 9/13/2021 7:11:12 PM
Error        :
```

3. After the VM is stopped, get a reference to either the OS or data disk attached to the VM with the `Get-AzDisk` cmdlet.

The following example selects the VM's OS disk.

```
$disk= Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $vm.StorageProfile.OsDisk.Name
```

The following example selects the VM's first data disk.

```
$disk= Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $vm.StorageProfile.DataDisks[0].Name
```

4. Now that you have a reference to the disk, set the size of the disk to 250 GiB.

## IMPORTANT

The new size should be greater than the existing disk size. The maximum allowed is 4,095 GiB for OS disks.

```
$disk.DiskSizeGB = 250
```

5. Next, update the disk image with the `Update-AzDisk` cmdlet.

```
Update-AzDisk `  
-ResourceGroupName $azResourceGroup `  
-Disk $disk -DiskName $disk.Name
```

The disk image is updated, and the output confirms the disk's new size.

```
ResourceGroupName      : myDemoResourceGroup  
ManagedBy             :  
/subscriptions/{GUID}/resourceGroups/myDemoResourceGroup/providers/Microsoft.Compute/virtualMachines/  
myDemoVM  
Sku                  : Microsoft.Azure.Management.Compute.Models.DiskSku  
TimeCreated           : 9/13/2021 6:41:10 PM  
CreationData          : Microsoft.Azure.Management.Compute.Models.CreationData  
DiskSizeGB            : 250  
DiskSizeBytes         : 268435456000  
UniqueId              : {GUID}  
ProvisioningState     : Succeeded  
DiskIOPSReadWrite     : 500  
DiskMBpsReadWrite    : 60  
DiskState              : Reserved  
Encryption            : Microsoft.Azure.Management.Compute.Models.Encryption  
Id                   :  
/subscriptions/{GUID}/resourceGroups/myDemoResourceGroup/providers/Microsoft.Compute/disks/myDemoData  
Disk  
Name                 : myDemoDataDisk  
Type                 : Microsoft.Compute/disks  
Location              : centralus
```

6. Finally, restart the VM with the `Start-AzVM` cmdlet.

```
Start-AzVM `  
-ResourceGroupName $azResourceGroup `  
-Name $azVMName
```

After a short pause, the output confirms that the machine is successfully started.

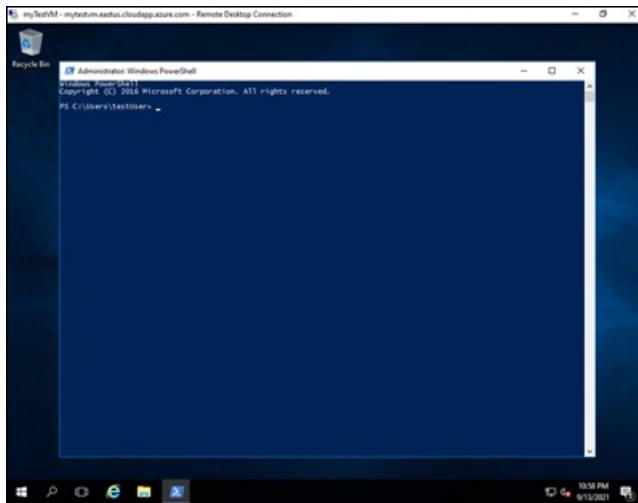
```
OperationId : abcd1234-ab12-cd34-123456abcdef  
Status      : Succeeded  
StartTime   : 9/13/2021 7:44:54 PM  
EndTime     : 9/13/2021 7:45:15 PM  
Error       :
```

## Expand the disk volume in the OS

Before you can take advantage of the new disk size, you need to expand the volume within the OS. Follow the steps below to expand the disk volume and take advantage of the new disk size.

1. Sign in to the [Azure portal](#).

- Locate the VM to which you've attached the data disk. Create a Remote Desktop Protocol (RDP) connection and sign in. If you no longer have access to an administrative account, create a credential object for a specified user name and password with the [Get-Credential](#) cmdlet.
- After you've established an RDP connection to the remote VM, select the Windows **Start** menu. Enter **PowerShell** in the search box and select **Windows PowerShell** to open a PowerShell window.



- Open PowerShell and run the following script. Change the value of the `-DriveLetter` variable as appropriate. For example, to resize the partition on the F: drive, use `$driveLetter = "F"`.

```
$driveLetter = "[Drive Letter]"
GetSize = (Get-PartitionSupportedSize -DriveLetter $driveLetter)
Resize-Partition ` 
    -DriveLetter $driveLetter ` 
    -Size $size.SizeMax
```

- Minimize the RDP window and switch back to Azure Cloud Shell. Use the [Get-AzDisk](#) cmdlet to verify that the disk was resized successfully.

```
Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup | Out-Host -Paging
```

## Upgrade a disk

There are several ways to respond to changes in your organization's workloads. For example, you may choose to upgrade a standard HDD to a premium SSD to handle increased demand.

Follow the steps in this section to upgrade a managed disk from standard to premium.

- Select the VM that contains the disk that you'll upgrade with the [Get-AzVM](#) cmdlet.

```
$vm = Get-AzVM ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName
```

- Before you can upgrade a VM's disk, you must stop the VM. Use the [Stop-AzVM](#) cmdlet to stop the VM. You'll be prompted for confirmation.

### IMPORTANT

Before you initiate a VM shutdown, always confirm that there are no important resources or data that could be lost.

```
Stop-AzVM ` 
    -ResourceGroupName $azResourceGroup ` 
    -Name $azVMName
```

After a short pause, the output confirms that the machine is successfully stopped.

```
OperationId : abcd1234-ab12-cd34-123456abcdef
Status       : Succeeded
StartTime    : 9/13/2021 7:10:23 PM
EndTime      : 9/13/2021 7:11:12 PM
Error        :
```

3. After the VM is stopped, get a reference to either the OS or data disk attached to the VM with the `Get-AzDisk` cmdlet.

The following example selects the VM's OS disk.

```
$disk= Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $vm.StorageProfile.OsDisk.Name
```

The following example selects the VM's first data disk.

```
$disk= Get-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -DiskName $vm.StorageProfile.DataDisks[0].Name
```

4. Now that you have a reference to the disk, set the disk's SKU to **Premium\_LRS**.

```
$disk.Sku = [Microsoft.Azure.Management.Compute.Models.DiskSku]::new('Premium_LRS')
```

5. Next, update the disk image with the `Update-AzDisk` cmdlet.

```
Update-AzDisk ` 
    -ResourceGroupName $azResourceGroup ` 
    -Disk $disk -DiskName $disk.Name
```

The disk image is updated. Use the following example code to validate that the disk's SKU has been upgraded.

```
$disk.Sku.Name
```

The output confirms the disk's new SKU.

```
Premium_LRS
```

6. Finally, restart the VM with the `Start-AzVM` cmdlet.

```
Start-AzVM ` 
-ResourceGroupName $azResourceGroup ` 
-Name $azVMName
```

After a short pause, the output confirms that the machine is successfully started.

```
OperationId : abcd1234-ab12-cd34-123456abcdef
Status       : Succeeded
StartTime    : 9/13/2021 7:44:54 PM
EndTime      : 9/13/2021 7:45:15 PM
Error        :
```

## Detach a data disk

You can detach a data disk from a VM when you want to attach it to a different VM, or when it's no longer needed. By default, detached disks are not deleted to prevent unintentional data loss. A detached disk will continue to incur storage charges until it's deleted.

1. First, select the VM to which the disk is attached with the `Get-AzVM` cmdlet.

```
$vm = Get-AzVM ` 
-ResourceGroupName $azResourceGroup ` 
-Name $azVMName
```

2. Next, detach the disk from the VM with the `Remove-AzVMDataDisk` cmdlet.

```
Remove-AzVMDataDisk ` 
-VM $vm ` 
-Name $azDataDiskName
```

3. Update the state of the VM with the `Update-AzVM` cmdlet to remove the data disk.

```
Update-AzVM ` 
-ResourceGroupName $azResourceGroup ` 
-VM $vm
```

After a short pause, the output confirms that the VM is successfully updated.

```
RequestId IsSuccess StatusCode ReasonPhrase
----- -----
True      OK      OK
```

## Delete a data disk

When you delete a VM, data disks attached to the VM remain provisioned and continue to incur charges until they're deleted. This default behavior helps prevent data loss caused by unintentional deletion.

You can use the following sample PowerShell script to delete unattached disks. The retrieval of disks is limited to the `myDemoResourceGroup` because the `-ResourceGroupName` switch is used with the `Get-AzDisk` cmdlet.

```

# Get all disks in resource group $azResourceGroup
$allDisks = Get-AzDisk -ResourceGroupName $azResourceGroup

# Determine the number of disks in the collection
if($allDisks.Count -ne 0) {

    Write-Host "Found $($allDisks.Count) disks."

    # Iterate through the collection
    foreach ($disk in $allDisks) {

        # Use the disk's "ManagedBy" property to determine if it is unattached
        if($disk.ManagedBy -eq $null) {

            # Confirm that the disk can be deleted
            Write-Host "Deleting unattached disk $($disk.Name)."
            $confirm = Read-Host "Continue? (Y/N)"
            if ($confirm.ToUpper() -ne 'Y') { break }
            else {

                # Delete the disk
                $disk | Remove-AzDisk -Force
                Write-Host "Unattached disk $($disk.Name) deleted."
            }
        }
    }
}

```

The unattached data disk is deleted as shown by the output.

```

Name      : abcd1234-ab12-cd34-ef56-abcdef123456
StartTime : 9/13/2021 10:14:05 AM
EndTime   : 9/13/2021 10:14:35 AM
Status    : Succeeded
Error     :

```

## Clean up resources

When no longer needed, delete the resource group, VM, and all related resources. You can use the following sample PowerShell script to delete the resource group created earlier in this tutorial.

**Caution**

Use caution when deleting a resource group. To avoid the loss of important data, always confirm that there are no important resources or data contained within the resource group before it is deleted.

```
Remove-AzResourceGroup -Name $azResourceGroup
```

You're prompted for confirmation. After a short pause, the `True` response confirms that the `myDemoResourceGroup` is successfully deleted.

```

Confirm
Are you sure you want to remove resource group 'myDemoResourceGroup'
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
True

```

## Next steps

In this tutorial, you learned how to:

- Create, attach, and initialize a data disk
- Verify a disk's status
- Initialize a disk
- Expand and upgrade a disk
- Detach and delete a disk

Advance to the next tutorial to learn how to automate VM configuration.

[Automate VM configuration](#)

# Tutorial - Deploy applications to a Windows virtual machine in Azure with the Custom Script Extension

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Window ✓ Flexible scale sets ✓ Uniform scale sets

To configure virtual machines (VMs) in a quick and consistent manner, you can use the [Custom Script Extension for Windows](#). In this tutorial you learn how to:

- Use the Custom Script Extension to install IIS
- Create a VM that uses the Custom Script Extension
- View a running IIS site after the extension is applied

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Custom script extension overview

The Custom Script Extension downloads and executes scripts on Azure VMs. This extension is useful for post deployment configuration, software installation, or any other configuration / management task. Scripts can be downloaded from Azure storage or GitHub, or provided to the Azure portal at extension run time.

The Custom Script extension integrates with Azure Resource Manager templates, and can also be run using the Azure CLI, PowerShell, Azure portal, or the Azure Virtual Machine REST API.

You can use the Custom Script Extension with both Windows and Linux VMs.

## Create virtual machine

Set the administrator username and password for the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Now you can create the VM with [New-AzVM](#). The following example creates a VM named *myVM* in the *EastUS* location. If they do not already exist, the resource group *myResourceGroupAutomate* and supporting network resources are created. To allow web traffic, the cmdlet also opens port *80*.

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroupAutomate" ` 
    -Name "myVM" ` 
    -Location "East US" ` 
    -VirtualNetworkName "myVnet" ` 
    -SubnetName "mySubnet" ` 
    -SecurityGroupName "myNetworkSecurityGroup" ` 
    -PublicIpAddressName "myPublicIpAddress" ` 
    -OpenPorts 80 ` 
    -Credential $cred
```

It takes a few minutes for the resources and VM to be created.

## Automate IIS install

Use [Set-AzVMExtension](#) to install the Custom Script Extension. The extension runs

```
powershell Add-WindowsFeature Web-Server
```

 to install the IIS webserver and then updates the *Default.htm* page to show the hostname of the VM:

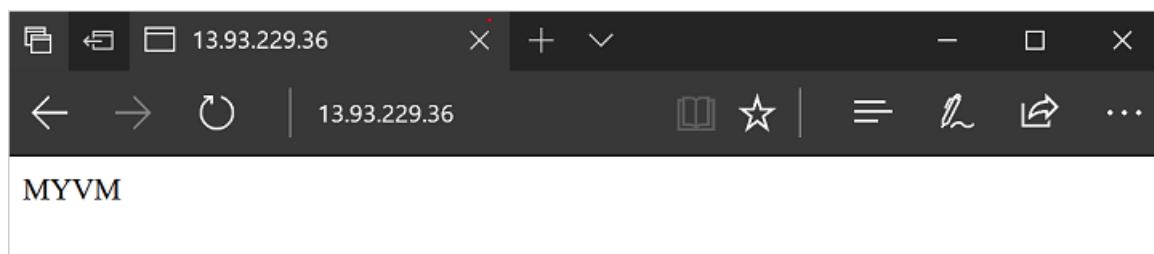
```
Set-AzVMExtension -ResourceGroupName "myResourceGroupAutomate" ` 
    -ExtensionName "IIS" ` 
    -VMName "myVM" ` 
    -Location "EastUS" ` 
    -Publisher Microsoft.Compute ` 
    -ExtensionType CustomScriptExtension ` 
    -TypeHandlerVersion 1.8 ` 
    -SettingString '{"commandToExecute":"powershell Add-WindowsFeature Web-Server; powershell Add-Content -Path \"C:\\inetpub\\wwwroot\\Default.htm\" -Value $($env:computername)"}'
```

## Test web site

Obtain the public IP address of your load balancer with [Get-AzPublicIPAddress](#). The following example obtains the IP address for *myPublicIPAddress* created earlier:

```
Get-AzPublicIPAddress ` 
    -ResourceGroupName "myResourceGroupAutomate" ` 
    -Name "myPublicIPAddress" | select IpAddress
```

You can then enter the public IP address in to a web browser. The website is displayed, including the hostname of the VM that the load balancer distributed traffic to as in the following example:



## Next steps

In this tutorial, you automated the IIS install on a VM. You learned how to:

- Use the Custom Script Extension to install IIS
- Create a VM that uses the Custom Script Extension
- View a running IIS site after the extension is applied

Advance to the next tutorial to learn how to create custom VM images.

[Create custom VM images](#)

# Tutorial: Create Windows VM images with Azure PowerShell

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Images can be used to bootstrap deployments and ensure consistency across multiple VMs. In this tutorial, you create your own specialized image of an Azure virtual machine using PowerShell and store it in an Azure Compute Gallery (formerly known as Shared Image Gallery). You learn how to:

- Create an Azure Compute Gallery
- Create an image definition
- Create an image version
- Create a VM from an image
- Share a gallery

## Before you begin

The steps below detail how to take an existing VM and turn it into a re-usable custom image that you can use to create new VMs.

To complete the example in this tutorial, you must have an existing virtual machine. If needed, you can see the [PowerShell quickstart](#) to create a VM to use for this tutorial. When working through the tutorial, replace the resource names where needed.

## Overview

An [Azure Compute Gallery](#) simplifies custom image sharing across your organization. Custom images are like marketplace images, but you create them yourself. Custom images can be used to bootstrap configurations such as preloading applications, application configurations, and other OS configurations.

The Azure Compute Gallery lets you share your custom VM images with others. Choose which images you want to share, which regions you want to make them available in, and who you want to share them with.

The Azure Compute Gallery feature has multiple resource types:

RESOURCE	DESCRIPTION
Image source	This is a resource that can be used to create an <a href="#">image version</a> in a gallery. An image source can be an existing Azure VM that is either <a href="#">generalized or specialized</a> , a managed image, a snapshot, or an image version in another gallery.
Gallery	Like the Azure Marketplace, a <a href="#">gallery</a> is a repository for managing and sharing images and <a href="#">VM applications</a> , but you control who has access.

RESOURCE	DESCRIPTION
Image definition	Image definitions are created within a gallery and carry information about the image and requirements for using it internally. This includes whether the image is Windows or Linux, release notes, and minimum and maximum memory requirements. It is a definition of a type of image.
Image version	An <b>image version</b> is what you use to create a VM when using a gallery. You can have multiple versions of an image as needed for your environment. Like a managed image, when you use an <b>image version</b> to create a VM, the image version is used to create new disks for the VM. Image versions can be used multiple times.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Get the VM

You can see a list of VMs that are available in a resource group using [Get-AzVM](#). Once you know the VM name and what resource group, you can use `Get-AzVM` again to get the VM object and store it in a variable to use later. This example gets an VM named *sourceVM* from the "myResourceGroup" resource group and assigns it to the variable `$sourceVM`.

```
$sourceVM = Get-AzVM ` 
-Name sourceVM ` 
-ResourceGroupName myResourceGroup
```

## Create a resource group

Create a resource group with the [New-AzResourceGroup](#) command.

An Azure resource group is a logical container into which Azure resources are deployed and managed. In the following example, a resource group named *myGalleryRG* is created in the *EastUS* region:

```
$resourceGroup = New-AzResourceGroup ` 
-Name 'myGalleryRG' ` 
-Location 'EastUS'
```

## Create a gallery

A gallery is the primary resource used for enabling image sharing. Allowed characters for gallery name are uppercase or lowercase letters, digits, dots, and periods. The gallery name cannot contain dashes. Gallery names must be unique within your subscription.

Create a gallery using [New-AzGallery](#). The following example creates a gallery named *myGallery* in the *myGalleryRG* resource group.

```
$gallery = New-AzGallery ` 
    -GalleryName 'myGallery' ` 
    -ResourceGroupName $resourceGroup.ResourceGroupName ` 
    -Location $resourceGroup.Location ` 
    -Description 'Azure Compute Gallery for my organization'
```

## Create an image definition

Image definitions create a logical grouping for images. They are used to manage information about the image versions that are created within them. Image definition names can be made up of uppercase or lowercase letters, digits, dots, dashes and periods. For more information about the values you can specify for an image definition, see [Image definitions](#).

Create the image definition using [New-AzGalleryImageDefinition](#). In this example, the gallery image is named `myGalleryImage` and is created for a specialized image.

```
$galleryImage = New-AzGalleryImageDefinition ` 
    -GalleryName $gallery.Name ` 
    -ResourceGroupName $resourceGroup.ResourceGroupName ` 
    -Location $gallery.Location ` 
    -Name 'myImageDefinition' ` 
    -OsState specialized ` 
    -OsType Windows ` 
    -Publisher 'myPublisher' ` 
    -Offer 'myOffer' ` 
    -Sku 'mySKU'
```

## Create an image version

Create an image version from a VM using [New-AzGalleryImageVersion](#).

Allowed characters for image version are numbers and periods. Numbers must be within the range of a 32-bit integer. Format: *MajorVersion.MinorVersion.Patch*.

In this example, the image version is `1.0.0` and it's replicated to both *East US* and *South Central US* datacenters. When choosing target regions for replication, you need to include the *source* region as a target for replication.

To create an image version from the VM, use `$vm.Id.ToString()` for the `-Source`.

```
$region1 = @{Name='South Central US';ReplicaCount=1} 
$region2 = @{Name='East US';ReplicaCount=2} 
$targetRegions = @($region1,$region2)

New-AzGalleryImageVersion ` 
    -GalleryImageDefinitionName $galleryImage.Name ` 
    -GalleryImageVersionName '1.0.0' ` 
    -GalleryName $gallery.Name ` 
    -ResourceGroupName $resourceGroup.ResourceGroupName ` 
    -Location $resourceGroup.Location ` 
    -TargetRegion $targetRegions ` 
    -Source $sourceVM.Id.ToString() ` 
    -PublishingProfileEndOfLifeDate '2030-12-01'
```

It can take a while to replicate the image to all of the target regions.

## Create a VM

Once you have a specialized image, you can create one or more new VMs. Using the [New-AzVM](#) cmdlet. To use

the image, use `Set-AzVMSourceImage` and set the `-Id` to the image definition ID (`$galleryImage.Id` in this case) to always use the latest image version.

Replace resource names as needed in this example.

```
# Create some variables for the new VM.
$resourceGroup = "myResourceGroup"
$location = "South Central US"
$vmName = "mySpecializedVM"

# Create a resource group
New-AzResourceGroup -Name $resourceGroup -Location $location

# Create the network resources.
$subnetConfig = New-AzVirtualNetworkSubnetConfig -Name mySubnet -AddressPrefix 192.168.1.0/24
$vnet = New-AzVirtualNetwork -ResourceGroupName $resourceGroup -Location $location ` 
    -Name MYvNET -AddressPrefix 192.168.0.0/16 -Subnet $subnetConfig
$pip = New-AzPublicIpAddress -ResourceGroupName $resourceGroup -Location $location ` 
    -Name "mypublicdns$(Get-Random)" -AllocationMethod Static -IdleTimeoutInMinutes 4
$nsgRuleRDP = New-AzNetworkSecurityRuleConfig -Name myNetworkSecurityGroupRuleRDP -Protocol Tcp ` 
    -Direction Inbound -Priority 1000 -SourceAddressPrefix * -SourcePortRange * -DestinationAddressPrefix * ` 
    -DestinationPortRange 3389 -Access Deny
$nsg = New-AzNetworkSecurityGroup -ResourceGroupName $resourceGroup -Location $location ` 
    -Name myNetworkSecurityGroup -SecurityRules $nsgRuleRDP
$nic = New-AzNetworkInterface -Name $vmName -ResourceGroupName $resourceGroup -Location $location ` 
    -SubnetId $vnet.Subnets[0].Id -PublicIpAddressId $pip.Id -NetworkSecurityGroupId $nsg.Id

# Create a virtual machine configuration using $imageVersion.Id to specify the image version.
$vmConfig = New-AzVMConfig -VMName $vmName -VMSize Standard_D1_v2 | ` 
    Set-AzVMSourceImage -Id $galleryImage.Id | ` 
    Add-AzVMNetworkInterface -Id $nic.Id

# Create a virtual machine
New-AzVM -ResourceGroupName $resourceGroup -Location $location -VM $vmConfig
```

## Share the gallery

We recommend that you share access at the gallery level. Use an email address and the [Get-AzADUser](#) cmdlet to get the object ID for the user, then use [New-AzRoleAssignment](#) to give them access to the gallery. Replace the example email, `alinne_montes@contoso.com` in this example, with your own information.

```
# Get the object ID for the user
$user = Get-AzADUser -StartsWith alinne_montes@contoso.com
# Grant access to the user for our gallery
New-AzRoleAssignment ` 
    -ObjectId $user.Id ` 
    -RoleDefinitionName Reader ` 
    -ResourceName $gallery.Name ` 
    -ResourceType Microsoft.Compute/galleries ` 
    -ResourceGroupName $resourceGroup.ResourceGroupName
```

## Clean up resources

When no longer needed, you can use the [Remove-AzResourceGroup](#) cmdlet to remove the resource group, and all related resources:

```
# Delete the gallery  
Remove-AzResourceGroup -Name myGalleryRG  
  
# Delete the VM  
Remove-AzResourceGroup -Name myResourceGroup
```

## Azure Image Builder

Azure also offers a service, built on Packer, [Azure VM Image Builder](#). Simply describe your customizations in a template, and it will handle the image creation.

## Next steps

In this tutorial, you created a specialized VM image. You learned how to:

- Create an Azure Compute Gallery
- Create an image definition
- Create an image version
- Create a VM from an image
- Share a gallery

Advance to the next tutorial to learn about how to create highly available virtual machines.

[Create highly available VMs](#)

# Create and deploy virtual machines in an availability set using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this tutorial, you learn how to increase the availability and reliability of your Virtual Machines (VMs) using Availability Sets. Availability Sets make sure the VMs you deploy on Azure are distributed across multiple, isolated hardware nodes, in a cluster.

In this tutorial, you learn how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes
- Check Azure Advisor

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create an availability set

The hardware in a location is divided into multiple update domains and fault domains. An **update domain** is a group of VMs and underlying physical hardware that can be rebooted at the same time. VMs in the same **fault domain** share common storage as well as a common power source and network switch.

You can create an availability set using [New-AzAvailabilitySet](#). In this example, the number of both update and fault domains is 2 and the availability set is named *myAvailabilitySet*.

Create a resource group.

```
New-AzResourceGroup ` 
-Name myResourceGroupAvailability ` 
-Location EastUS
```

Create a managed availability set using [New-AzAvailabilitySet](#) with the `-sku aligned` parameter.

```
New-AzAvailabilitySet ` 
-Location "EastUS" ` 
-Name "myAvailabilitySet" ` 
-ResourceGroupName "myResourceGroupAvailability" ` 
-Sku aligned ` 
-PlatformFaultDomainCount 2 ` 
-PlatformUpdateDomainCount 2
```

# Create VMs inside an availability set

VMs must be created within the availability set to make sure they're correctly distributed across the hardware. You can't add an existing VM to an availability set after it's created.

When you create a VM with [New-AzVM](#), you use the `-AvailabilitySetName` parameter to specify the name of the availability set.

First, set an administrator username and password for the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Now create two VMs with [New-AzVM](#) in the availability set.

```
for ($i=1; $i -le 2; $i++)
{
    New-AzVm ` 
        -ResourceGroupName "myResourceGroupAvailability" ` 
        -Name "myVM$i" ` 
        -Location "East US" ` 
        -VirtualNetworkName "myVnet" ` 
        -SubnetName "mySubnet" ` 
        -SecurityGroupName "myNetworkSecurityGroup" ` 
        -PublicIpAddressName "myPublicIpAddress$i" ` 
        -AvailabilitySetName "myAvailabilitySet" ` 
        -Credential $cred
}
```

It takes a few minutes to create and configure both VMs. When finished, you have two virtual machines distributed across the underlying hardware.

If you look at the availability set in the portal by going to **Resource Groups > myResourceGroupAvailability > myAvailabilitySet**, you should see how the VMs are distributed across the two fault and update domains.

The screenshot shows the Azure portal interface for managing an availability set. The top navigation bar includes 'Microsoft Azure', 'Resource groups', 'myResourceGroupAvailability', 'myAvailabilitySet', and a search bar. The main content area displays the 'myAvailabilitySet' details under the 'Essentials' tab. It lists the resource group as 'myResourceGroupAvailability', location as 'East US', subscription name as 'Azure', and subscription ID as '<Subscription ID>'. It also indicates there are 2 fault domains and 2 update domains, with 2 virtual machines (myVM1 and myVM2) managed by the set. Below this, a table shows the status of the two VMs: myVM1 is running in Fault Domain 0 and Update Domain 0, while myVM2 is also running in Fault Domain 1 and Update Domain 1.

NAME	STATUS	FAULT DOMAIN	UPDATE DOMAIN
myVM1	Running	0	0
myVM2	Running	1	1

#### NOTE

Under certain circumstances, 2 VMs in the same AvailabilitySet could share the same FaultDomain. This can be confirmed by going into your availability set and checking the Fault Domain column. This can be caused by the following sequence of events while deploying the VMs:

1. The 1st VM is Deployed
2. The 1st VM is Stopped/Deallocated
3. The 2nd VM is Deployed. Under these circumstances, the OS Disk of the 2nd VM might be created on the same Fault Domain as the 1st VM, and so the 2nd VM will also land on the same FaultDomain. To avoid this issue, it's recommended to not stop/deallocate the VMs between deployments.

## Check for available VM sizes

When you create a VM inside a availability set, you need to know what VM sizes are available on the hardware. Use [Get-AzVMSize](#) command to get all available sizes for virtual machines that you can deploy in the availability set.

```
Get-AzVMSize ` 
    -ResourceGroupName "myResourceGroupAvailability" ` 
    -AvailabilitySetName "myAvailabilitySet"
```

## Check Azure Advisor

You can also use Azure Advisor to get more information on how to improve the availability of your VMs. Azure Advisor analyzes your configuration and usage telemetry, then recommends solutions that can help you improve the cost effectiveness, performance, availability, and security of your Azure resources.

Sign in to the [Azure portal](#), select **All services**, and type **Advisor**. The Advisor dashboard shows personalized recommendations for the selected subscription. For more information, see [Get started with Azure Advisor](#).

## Next steps

In this tutorial, you learned how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes
- Check Azure Advisor

Advance to the next tutorial to learn about virtual machine scale sets.

[Create a VM scale set](#)

# Tutorial: Create a virtual machine scale set and deploy a highly available app on Windows

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Virtual machine scale sets with [Flexible orchestration](#) let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule.

In this tutorial, you deploy a virtual machine scale set in Azure and learn how to:

- Create a resource group.
- Create a Flexible scale set with a load balancer.
- Add IIS to the scale set instances using the [Run command](#).
- Open port 80 to HTTP traffic.
- Test the scale set.

## Scale Set overview

Scale sets provide the following key benefits:

- Easy to create and manage multiple VMs
- Provides high availability and application resiliency by distributing VMs across fault domains
- Allows your application to automatically scale as resource demand changes
- Works at large-scale

With Flexible orchestration, Azure provides a unified experience across the Azure VM ecosystem. Flexible orchestration offers high availability guarantees (up to 1000 VMs) by spreading VMs across fault domains in a region or within an Availability Zone. This enables you to scale out your application while maintaining fault domain isolation that is essential to run quorum-based or stateful workloads, including:

- Quorum-based workloads
- Open-source databases
- Stateful applications
- Services that require high availability and large scale
- Services that want to mix virtual machine types or leverage Spot and on-demand VMs together
- Existing Availability Set applications

Learn more about the differences between Uniform scale sets and Flexible scale sets in [Orchestration Modes](#).

## Create a scale set

Use the Azure portal to create a Flexible scale set.

1. Open the [Azure portal](#).
2. Search for and select **Virtual machine scale sets**.
3. Select **Create** on the **Virtual machine scale sets** page. The **Create a virtual machine scale set** will open.
4. Select the subscription that you want to use for **Subscription**.

5. For **Resource group**, select **Create new** and type *myVMSSRG* for the name and then select **OK**.

#### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	myAzureSubscription
Resource group *	(New) myVMSS_group
<a href="#">Create new</a>	

6. For **Virtual machine scale set name**, type *myVMSS*.

7. For **Region**, select a region that is close to you like *East US*.

#### Scale set details

Virtual machine scale set name *	myVMSS
Region *	(US) East US
Availability zone ⓘ	None

8. Leave **Availability zone** as blank for this example.

9. For **Orchestration mode**, select **Flexible**.

10. Leave the default of **1** for fault domain count or choose another value from the drop-down.

#### Orchestration

A scale set has a "scale set model" that defines the attributes of virtual machine instances (size, number of data disks, etc). As the number of instances in the scale set changes, new instances are added based on the scale set model.

[Learn more about the scale set model ↗](#)

Orchestration mode \* ⓘ

- Uniform**: optimized for large scale stateless workloads with identical instances  
 **Flexible (preview)**: achieve high availability at scale with identical or multiple virtual machine types

**!** This virtual machine scale set will be created as a scale set with flexible orchestration mode (preview). To enable the preview, you must register your subscription. [Learn more ↗](#)

Fault domain count \* ⓘ

1

11. For **Image**, select *Windows Server 2019 Datacenter - Gen 1*.

12. For **Size**, leave the default value or select a size like *Standard\_E2s\_V3*.

13. For **Username**, type the name to use for the administrator account, like *azureuser*.

14. In **Password** and **Confirm password**, type a strong password for the administrator account.

15. On the **Networking** tab, under **Load balancing**, select **Use a load balancer**.

16. For **Load balancing options**, leave the default of **Azure load balancer**.

17. For **Select a load balancer**, select **Create new**.

## Load balancing

You can place this virtual machine scale set in the backend pool of an existing Azure load balancing solution. [Learn more](#)

Use a load balancer

### Load balancing settings

- **Application Gateway** is an HTTP/HTTPS web traffic load balancer with URL-based routing, SSL termination, session persistence, and web application firewall. [Learn more about Application Gateway](#)
- **Azure Load Balancer** supports all TCP/UDP network traffic, port-forwarding, and outbound flows. [Learn more about Azure Load Balancer](#)

Load balancing options * ⓘ	Azure load balancer
Select a load balancer * ⓘ	(new) myVMSS2021-lb
	<a href="#">Create new</a>
Select a backend pool * ⓘ	(new) bepool
	<a href="#">Create new</a>

18. On the **Create a load balancer** page, type in a name for your load balancer and **Public IP address name**.
19. For **Domain name label**, type in a name to use as a prefix for your domain name. This name must be unique.
20. When you are done, select **Create**.

### Create a load balancer

Azure Load Balancer enables you to scale your applications and create high availability for your services. Load Balancer supports inbound and outbound scenarios, provides low latency and high throughput, and scales up to millions of flows for all TCP and UDP applications. [Learn more about Azure Load Balancer](#)

Your load balancer will be placed in the same subscription, resource group, and region as your virtual machine scale set. Azure will configure basic settings for the frontend IP, backend address pools, NAT rules, and NAT pools for this load balancer automatically.

Name * ⓘ	myVMSS-lb
Public IP address name * ⓘ	myVMSS-ip
Domain name label ⓘ	myvmss2021.eastus.cloudapp.azure.com
SKU	Standard
Type	Public
Availability zone ⓘ	Zone-redundant

**Create** **Discard**

21. Back on the **Networking** tab, leave the default name for the backend pool.
22. On the **Scaling** tab, leave the default instance count as 2, or add in your own value. This is the number of VMs that will be created, so be aware of the costs and the limits on your subscription if you change this value.
23. Leave the **Scaling policy** set to *Manual*.

An Azure virtual machine scale set can automatically increase or decrease the number of VM instances that run your application. This automated and elastic behavior reduces the management overhead to monitor and optimize the performance of your application. [Learn more about VMSS scaling](#)

Initial instance count \*

**Scaling**

Scaling policy [\(i\)](#)

Manual  
 Custom

**Scale-In policy**

Configure the order in which virtual machines are selected for deletion during a scale-in operation.  
[Learn more about scale-in policies](#)

Scale-in policy  [\(i\)](#)

**Info** Scale-in policy is not supported for virtual machine scale sets with flexible orchestration mode (preview).

24. When you are done, select **Review + create**.
25. Once you see that validation has passed, you can select **Create** at the bottom of the page to deploy your scale set.
26. When the deployment is complete, select **Go to resource** to see your scale set.

## View the VMs in your scale set

On the page for the scale set, select **Instances** from the left menu.

You will see a list of VMs that are part of your scale set. This list includes:

- The name of the VM
- The computer name used by the VM.
- The current status of the VM, like *Running*.
- The *Provisioning state* of the VM, like *Succeeded*.

Name	Computer name	Status	Provisioning state
myVMSS_351a2d68	myvmss6nEQF90O	Running	Succeeded
myVMSS_c4dac4c9	myvmss6nGESCAF	Running	Succeeded

## Enable IIS using RunCommand

To test the scale-set, we can enable IIS on each of the VMs using the [Run Command](#).

1. Select the first VM in the list of **Instances**.
2. In the left menu, under **Operations**, select **Run command**. The **Run command** page will open.
3. Select **RunPowerShellScript** from the list of commands. The **Run Command Script** page will open.
4. Under **PowerShell Script**, paste in the following snippet:

```
Add-WindowsFeature Web-Server  
Set-Content -Path "C:\inetpub\wwwroot\Default.htm" -Value "Hello world from host $($env:computername)  
!"
```

5. When you are done, select **Run**. You will see the progress in the **Output** window.
6. Once the script is complete on the first VM, you can select the X in the upper-right to close the page.
7. Go back to your list of scale set instances and use the **Run command** on each VM in the scale set.

## Open port 80

Open port 80 on your scale set by adding an inbound rule to your network security group (NSG).

1. On the page for your scale set, select **Networking** from the left menu. The **Networking** page will open.
2. Select **Add inbound port rule**. The **Add inbound security rule** page will open.
3. Under **Service**, select *HTTP* and then select **Add** at the bottom of the page.

## Test your scale set

Test your scale set by connecting to it from a browser.

1. On the **Overview** page for your scale set, copy the Public IP address.
2. Open another tab in your browser and paste the IP address into the address bar.
3. When the page loads, take a note of the compute name that is shown.
4. Refresh the page until you see the computer name change.

## Delete your scale set

When you are done, you should delete the resource group, which will delete everything you deployed for your scale set.

1. On the page for your scale set, select the **Resource group**. The page for your resource group will open.
2. At the top of the page, select **Delete resource group**.
3. In the **Are you sure you want to delete** page, type in the name of your resource group and then select **Delete**.

## Next steps

In this tutorial, you created a virtual machine scale set. You learned how to:

- Create a resource group.
- Create a Flexible scale set with a load balancer.
- Add IIS to the scale set instances using the **Run command**.
- Open port 80 to HTTP traffic.
- Test the scale set.

Advance to the next tutorial to learn more about load balancing concepts for virtual machines.

[Load balance virtual machines](#)

# Tutorial: Load balance Windows virtual machines in Azure to create a highly available application with Azure PowerShell

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Uniform scale sets

Load balancing provides a higher level of availability by spreading incoming requests across multiple virtual machines. In this tutorial, you learn about the different components of the Azure load balancer that distribute traffic and provide high availability. You learn how to:

- Create an Azure load balancer
- Create a load balancer health probe
- Create load balancer traffic rules
- Use the Custom Script Extension to create a basic IIS site
- Create virtual machines and attach to a load balancer
- View a load balancer in action
- Add and remove VMs from a load balancer

## Azure load balancer overview

An Azure load balancer is a Layer-4 (TCP, UDP) load balancer that provides high availability by distributing incoming traffic among healthy VMs. A load balancer health probe monitors a given port on each VM and only distributes traffic to an operational VM.

You define a front-end IP configuration that contains one or more public IP addresses. This front-end IP configuration allows your load balancer and applications to be accessible over the Internet.

Virtual machines connect to a load balancer using their virtual network interface card (NIC). To distribute traffic to the VMs, a back-end address pool contains the IP addresses of the virtual (NICs) connected to the load balancer.

To control the flow of traffic, you define load balancer rules for specific ports and protocols that map to your VMs.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create Azure load balancer

This section details how you can create and configure each component of the load balancer. Before you can create your load balancer, create a resource group with [New-AzResourceGroup](#). The following example creates a resource group named *myResourceGroupLoadBalancer* in the *EastUS* location:

```
New-AzResourceGroup ` 
-ResourceGroupName "myResourceGroupLoadBalancer" ` 
-Location "EastUS"
```

## Create a public IP address

To access your app on the Internet, you need a public IP address for the load balancer. Create a public IP address with [New-AzPublicIpAddress](#). The following example creates a public IP address named *myPublicIP* in the *myResourceGroupLoadBalancer* resource group:

```
$publicIP = New-AzPublicIpAddress ` 
-ResourceGroupName "myResourceGroupLoadBalancer" ` 
-Location "EastUS" ` 
-AllocationMethod "Static" ` 
-Name "myPublicIP"
```

## Create a load balancer

Create a frontend IP pool with [New-AzLoadBalancerFrontendIpConfig](#). The following example creates a frontend IP pool named *myFrontEndPool* and attaches the *myPublicIP* address:

```
$frontendIP = New-AzLoadBalancerFrontendIpConfig ` 
-Name "myFrontEndPool" ` 
-PublicIpAddress $publicIP
```

Create a backend address pool with [New-AzLoadBalancerBackendAddressPoolConfig](#). The VMs attach to this backend pool in the remaining steps. The following example creates a backend address pool named *myBackEndPool*:

```
$backendPool = New-AzLoadBalancerBackendAddressPoolConfig ` 
-Name "myBackEndPool"
```

Now, create the load balancer with [New-AzLoadBalancer](#). The following example creates a load balancer named *myLoadBalancer* using the frontend and backend IP pools created in the preceding steps:

```
$lb = New-AzLoadBalancer ` 
-ResourceGroupName "myResourceGroupLoadBalancer" ` 
-Name "myLoadBalancer" ` 
-Location "EastUS" ` 
-FrontendIpConfiguration $frontendIP ` 
-BackendAddressPool $backendPool
```

## Create a health probe

To allow the load balancer to monitor the status of your app, you use a health probe. The health probe dynamically adds or removes VMs from the load balancer rotation based on their response to health checks. By default, a VM is removed from the load balancer distribution after two consecutive failures at 15-second intervals. You create a health probe based on a protocol or a specific health check page for your app.

The following example creates a TCP probe. You can also create custom HTTP probes for more fine grained health checks. When using a custom HTTP probe, you must create the health check page, such as *healthcheck.aspx*. The probe must return an **HTTP 200 OK** response for the load balancer to keep the host in rotation.

To create a TCP health probe, you use [Add-AzLoadBalancerProbeConfig](#). The following example creates a health probe named *myHealthProbe* that monitors each VM on *TCP port 80*:

```
Add-AzLoadBalancerProbeConfig ` 
-Name "myHealthProbe" ` 
-LoadBalancer $lb ` 
-Protocol tcp ` 
-Port 80 ` 
-IntervalInSeconds 15 ` 
-ProbeCount 2
```

To apply the health probe, update the load balancer with [Set-AzLoadBalancer](#):

```
Set-AzLoadBalancer -LoadBalancer $lb
```

### Create a load balancer rule

A load balancer rule is used to define how traffic is distributed to the VMs. You define the front-end IP configuration for the incoming traffic and the back-end IP pool to receive the traffic, along with the required source and destination port. To make sure only healthy VMs receive traffic, you also define the health probe to use.

Create a load balancer rule with [Add-AzLoadBalancerRuleConfig](#). The following example creates a load balancer rule named *myLoadBalancerRule* and balances traffic on *TCP* port *80*:

```
$probe = Get-AzLoadBalancerProbeConfig -LoadBalancer $lb -Name "myHealthProbe"

Add-AzLoadBalancerRuleConfig ` 
-Name "myLoadBalancerRule" ` 
-LoadBalancer $lb ` 
-FrontendIpConfiguration $lb.FrontendIpConfigurations[0] ` 
-BackendAddressPool $lb.BackendAddressPools[0] ` 
-Protocol Tcp ` 
-FrontendPort 80 ` 
-BackendPort 80 ` 
-Probe $probe
```

Update the load balancer with [Set-AzLoadBalancer](#):

```
Set-AzLoadBalancer -LoadBalancer $lb
```

## Configure virtual network

Before you deploy some VMs and can test your balancer, create the supporting virtual network resources. For more information about virtual networks, see the [Manage Azure Virtual Networks](#) tutorial.

### Create network resources

Create a virtual network with [New-AzVirtualNetwork](#). The following example creates a virtual network named *myVnet* with *mySubnet*.

```

# Create subnet config
$subnetConfig = New-AzVirtualNetworkSubnetConfig ` 
    -Name "mySubnet" ` 
    -AddressPrefix 192.168.1.0/24

# Create the virtual network
$vnet = New-AzVirtualNetwork ` 
    -ResourceGroupName "myResourceGroupLoadBalancer" ` 
    -Location "EastUS" ` 
    -Name "myVnet" ` 
    -AddressPrefix 192.168.0.0/16 ` 
    -Subnet $subnetConfig

```

Virtual NICs are created with [New-AzNetworkInterface](#). The following example creates three virtual NICs. (One virtual NIC for each VM you create for your app in the following steps). You can create additional virtual NICs and VMs at any time and add them to the load balancer:

```

for ($i=1; $i -le 3; $i++)
{
    New-AzNetworkInterface ` 
        -ResourceGroupName "myResourceGroupLoadBalancer" ` 
        -Name myVM$i ` 
        -Location "EastUS" ` 
        -Subnet $vnet.Subnets[0] ` 
        -LoadBalancerBackendAddressPool $lb.BackendAddressPools[0]
}

```

## Create virtual machines

To improve the high availability of your app, place your VMs in an availability set.

Create an availability set with [New-AzAvailabilitySet](#). The following example creates an availability set named *myAvailabilitySet*.

```

$availabilitySet = New-AzAvailabilitySet ` 
    -ResourceGroupName "myResourceGroupLoadBalancer" ` 
    -Name "myAvailabilitySet" ` 
    -Location "EastUS" ` 
    -Sku aligned ` 
    -PlatformFaultDomainCount 2 ` 
    -PlatformUpdateDomainCount 2

```

Set an administrator username and password for the VMs with [Get-Credential](#):

```
$cred = Get-Credential
```

Now you can create the VMs with [New-AzVM](#). The following example creates three VMs and the required virtual network components if they do not already exist:

```

for ($i=1; $i -le 3; $i++)
{
    New-AzVm ` 
        -ResourceGroupName "myResourceGroupLoadBalancer" ` 
        -Name "myVM$i" ` 
        -Location "East US" ` 
        -VirtualNetworkName "myVnet" ` 
        -SubnetName "mySubnet" ` 
        -SecurityGroupName "myNetworkSecurityGroup" ` 
        -OpenPorts 80 ` 
        -AvailabilitySetName "myAvailabilitySet" ` 
        -Credential $cred ` 
        -AsJob
}

```

The `-AsJob` parameter creates the VM as a background task, so the PowerShell prompts return to you. You can view details of background jobs with the `Job` cmdlet. It takes a few minutes to create and configure all three VMs.

### Install IIS with Custom Script Extension

In a previous tutorial on [How to customize a Windows virtual machine](#), you learned how to automate VM customization with the Custom Script Extension for Windows. You can use the same approach to install and configure IIS on your VMs.

Use [Set-AzVMExtension](#) to install the Custom Script Extension. The extension runs

`powershell Add-WindowsFeature Web-Server` to install the IIS webserver and then updates the *Default.htm* page to show the hostname of the VM:

```

for ($i=1; $i -le 3; $i++)
{
    Set-AzVMExtension ` 
        -ResourceGroupName "myResourceGroupLoadBalancer" ` 
        -ExtensionName "IIS" ` 
        -VMName myVM$i ` 
        -Publisher Microsoft.Compute ` 
        -ExtensionType CustomScriptExtension ` 
        -TypeHandlerVersion 1.8 ` 
        -SettingString '{"commandToExecute":"powershell Add-WindowsFeature Web-Server; powershell Add-Content -Path \'C:\\inetpub\\wwwroot\\Default.htm\' -Value $($env:computername)"}' ` 
        -Location EastUS
}

```

## Test load balancer

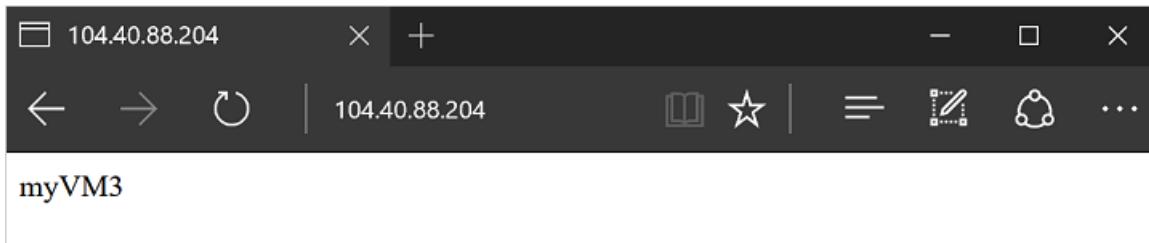
Obtain the public IP address of your load balancer with [Get-AzPublicIPAddress](#). The following example obtains the IP address for *myPublicIP* created earlier:

```

Get-AzPublicIPAddress ` 
    -ResourceGroupName "myResourceGroupLoadBalancer" ` 
    -Name "myPublicIP" | select IpAddress

```

You can then enter the public IP address in to a web browser. The website is displayed, including the hostname of the VM that the load balancer distributed traffic to as in the following example:



To see the load balancer distribute traffic across all three VMs running your app, you can force-refresh your web browser.

## Add and remove VMs

You may need to perform maintenance on the VMs running your app, such as installing OS updates. To deal with increased traffic to your app, you may need to add additional VMs. This section shows you how to remove or add a VM from the load balancer.

### Remove a VM from the load balancer

Get the network interface card with [Get-AzNetworkInterface](#), then set the *LoadBalancerBackendAddressPools* property of the virtual NIC to `$null`. Finally, update the virtual NIC:

```
$nic = Get-AzNetworkInterface `  
    -ResourceGroupName "myResourceGroupLoadBalancer" `  
    -Name "myVM2"  
$nic.IpConfigurations[0].LoadBalancerBackendAddressPools=$null  
Set-AzNetworkInterface -NetworkInterface $nic
```

To see the load balancer distribute traffic across the remaining two VMs running your app you can force-refresh your web browser. You can now perform maintenance on the VM, such as installing OS updates or performing a VM reboot.

### Add a VM to the load balancer

After performing VM maintenance, or if you need to expand capacity, set the *LoadBalancerBackendAddressPools* property of the virtual NIC to the *BackendAddressPool* from [Get-AzLoadBalancer](#):

Get the load balancer:

```
$lb = Get-AzLoadBalancer `  
    -ResourceGroupName myResourceGroupLoadBalancer `  
    -Name myLoadBalancer  
$nic.IpConfigurations[0].LoadBalancerBackendAddressPools=$lb.BackendAddressPools[0]  
Set-AzNetworkInterface -NetworkInterface $nic
```

## Next steps

In this tutorial, you created a load balancer and attached VMs to it. You learned how to:

- Create an Azure load balancer
- Create a load balancer health probe
- Create load balancer traffic rules
- Use the Custom Script Extension to create a basic IIS site
- Create virtual machines and attach to a load balancer
- View a load balancer in action
- Add and remove VMs from a load balancer

Advance to the next tutorial to learn how to manage VM networking.

[Manage VMs and virtual networks](#)

# Tutorial: Create and manage Azure virtual networks for Windows virtual machines with Azure PowerShell

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

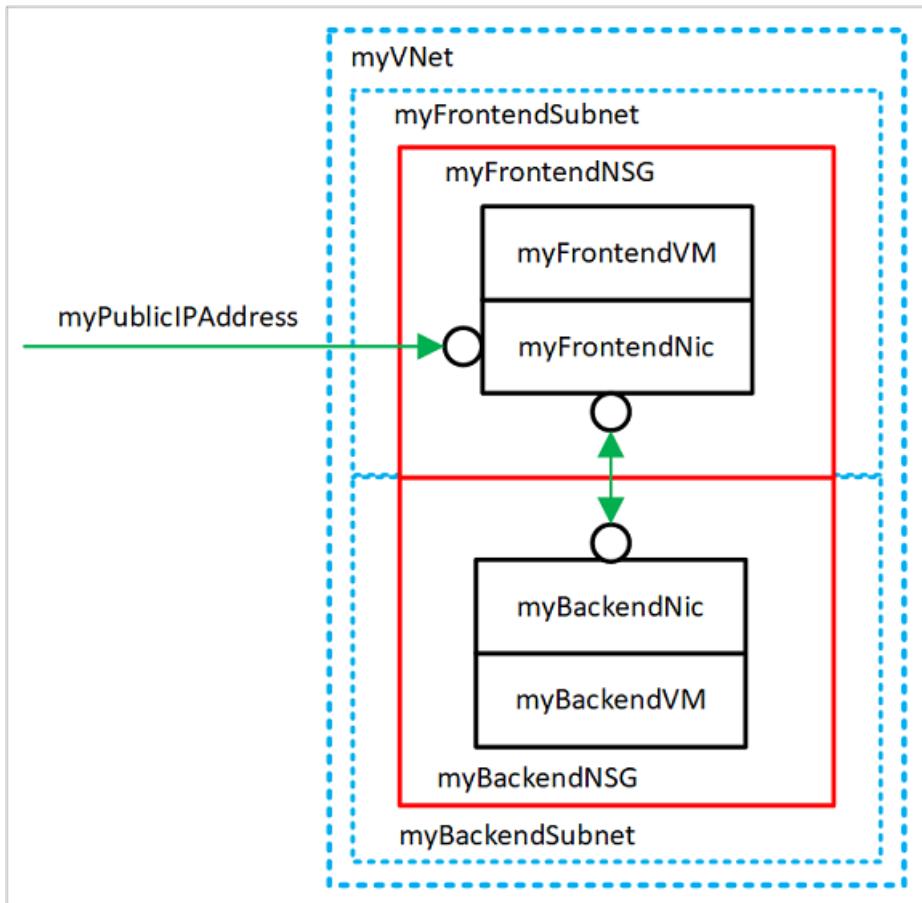
Azure virtual machines use Azure networking for internal and external network communication. This tutorial walks through deploying two virtual machines and configuring Azure networking for these VMs. The examples in this tutorial assume that the VMs are hosting a web application with a database back-end, however an application isn't deployed in the tutorial. In this tutorial, you learn how to:

- Create a virtual network and subnet
- Create a public IP address
- Create a front-end VM
- Secure network traffic
- Create back-end VM

## VM networking overview

Azure virtual networks enable secure network connections between virtual machines, the internet, and other Azure services such as Azure SQL Database. Virtual networks are broken down into logical segments called subnets. Subnets are used to control network flow, and as a security boundary. When deploying a VM, it generally includes a virtual network interface, which is attached to a subnet.

While completing this tutorial, you can see these resources created:



- `myVNet` - The virtual network that the VMs use to communicate with each other and the internet.
- `myFrontendSubnet` - The subnet in `myVNet` used by the front-end resources.
- `myPublicIPAddress` - The public IP address used to access `myFrontendVM` from the internet.
- `myFrontendNic` - The network interface used by `myFrontendVM` to communicate with `myBackendVM`.
- `myFrontendVM` - The VM used to communicate between the internet and `myBackendVM`.
- `myBackendNSG` - The network security group that controls communication between the `myFrontendVM` and `myBackendVM`.
- `myBackendSubnet` - The subnet associated with `myBackendNSG` and used by the back-end resources.
- `myBackendNic` - The network interface used by `myBackendVM` to communicate with `myFrontendVM`.
- `myBackendVM` - The VM that uses port 1433 to communicate with `myFrontendVM`.

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create subnet

For this tutorial, a single virtual network is created with two subnets. A front-end subnet for hosting a web application, and a back-end subnet for hosting a database server.

Before you can create a virtual network, create a resource group using [New-AzResourceGroup](#). The following example creates a resource group named `myRGNetwork` in the `EastUS` location:

```
New-AzResourceGroup -ResourceGroupName myRGNetwork -Location EastUS
```

Create a subnet configuration named *myFrontendSubnet* using [New-AzVirtualNetworkSubnetConfig](#):

```
$frontendSubnet = New-AzVirtualNetworkSubnetConfig `  
-Name myFrontendSubnet `  
-AddressPrefix 10.0.0.0/24
```

And, create a subnet configuration named *myBackendSubnet*:

```
$backendSubnet = New-AzVirtualNetworkSubnetConfig `  
-Name myBackendSubnet `  
-AddressPrefix 10.0.1.0/24
```

## Create virtual network

Create a VNET named *myVNet* using *myFrontendSubnet* and *myBackendSubnet* using [New-AzVirtualNetwork](#):

```
$vnet = New-AzVirtualNetwork `  
-ResourceGroupName myRGNetwork `  
-Location EastUS `  
-Name myVNet `  
-AddressPrefix 10.0.0.0/16 `  
-Subnet $frontendSubnet, $backendSubnet
```

At this point, a network has been created and segmented into two subnets, one for front-end services, and another for back-end services. In the next section, virtual machines are created and connected to these subnets.

## Create a public IP address

A public IP address allows Azure resources to be accessible on the internet. The allocation method of the public IP address can be configured as dynamic or static. By default, a public IP address is dynamically allocated. Dynamic IP addresses are released when a VM is deallocated. This behavior causes the IP address to change during any operation that includes a VM deallocation.

The allocation method can be set to static, which makes sure that the IP address stays assigned to a VM, even during a deallocated state. If you are using a static IP address, the IP address itself can't be specified. Instead, it's allocated from a pool of available addresses.

Create a public IP address named *myPublicIPAddress* using [New-AzPublicIpAddress](#):

```
$pip = New-AzPublicIpAddress `  
-ResourceGroupName myRGNetwork `  
-Location EastUS `  
-AllocationMethod Dynamic `  
-Name myPublicIPAddress
```

You could change the *-AllocationMethod* parameter to `static` to assign a static public IP address.

## Create a front-end VM

For a VM to communicate in a virtual network, it needs a virtual network interface (NIC). Create a NIC using [New-AzNetworkInterface](#):

```
$frontendNic = New-AzNetworkInterface `  
    -ResourceGroupName myRGNetwork `  
    -Location EastUS `  
    -Name myFrontend `  
    -SubnetId $vnet.Subnets[0].Id `  
    -PublicIpAddressId $pip.Id
```

Set the username and password needed for the administrator account on the VM using [Get-Credential](#). You use these credentials to connect to the VM in additional steps:

```
$cred = Get-Credential
```

Create the VMs using [New-AzVM](#).

```
New-AzVM `  
    -Credential $cred `  
    -Name myFrontend `  
    -PublicIpAddressName myPublicIPAddress `  
    -ResourceGroupName myRGNetwork `  
    -Location "EastUS" `  
    -Size Standard_D1 `  
    -SubnetName myFrontendSubnet `  
    -VirtualNetworkName myVNet
```

## Secure network traffic

A network security group (NSG) contains a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). NSGs can be associated to subnets or individual network interfaces. An NSG that is associated with a network interface only applies to the associated VM. When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet.

### Network security group rules

NSG rules define networking ports over which traffic is allowed or denied. The rules can include source and destination IP address ranges so that traffic is controlled between specific systems or subnets. NSG rules also include a priority (between 1—and 4096). Rules are evaluated in the order of priority. A rule with a priority of 100 is evaluated before a rule with priority 200.

All NSGs contain a set of default rules. The default rules can't be deleted, but because they are assigned the lowest priority, they can be overridden by the rules that you create.

- **Virtual network** - Traffic originating and ending in a virtual network is allowed both in inbound and outbound directions.
- **Internet** - Outbound traffic is allowed, but inbound traffic is blocked.
- **Load balancer** - Allow Azure's load balancer to probe the health of your VMs and role instances. If you are not using a load balanced set, you can override this rule.

### Create network security groups

Create an inbound rule named *myFrontendNSGRule* to allow incoming web traffic on *myFrontendVM* using [New-AzNetworkSecurityRuleConfig](#):

```
$nsgFrontendRule = New-AzNetworkSecurityRuleConfig ` 
-Name myFrontendNSGRule ` 
-Protocol Tcp ` 
-Direction Inbound ` 
-Priority 200 ` 
-SourceAddressPrefix * ` 
-SourcePortRange * ` 
-DestinationAddressPrefix * ` 
-DestinationPortRange 80 ` 
-Access Allow
```

You can limit internal traffic to *myBackendVM* from only *myFrontendVM* by creating an NSG for the back-end subnet. The following example creates an NSG rule named *myBackendNSGRule*.

```
$nsgBackendRule = New-AzNetworkSecurityRuleConfig ` 
-Name myBackendNSGRule ` 
-Protocol Tcp ` 
-Direction Inbound ` 
-Priority 100 ` 
-SourceAddressPrefix 10.0.0.0/24 ` 
-SourcePortRange * ` 
-DestinationAddressPrefix * ` 
-DestinationPortRange 1433 ` 
-Access Allow
```

Add a network security group named *myFrontendNSG* using [New-AzNetworkSecurityGroup](#):

```
$nsgFrontend = New-AzNetworkSecurityGroup ` 
-ResourceGroupName myRGNetwork ` 
-Location EastUS ` 
-Name myFrontendNSG ` 
-SecurityRules $nsgFrontendRule
```

Now, add a network security group named *myBackendNSG* using [New-AzNetworkSecurityGroup](#):

```
$nsgBackend = New-AzNetworkSecurityGroup ` 
-ResourceGroupName myRGNetwork ` 
-Location EastUS ` 
-Name myBackendNSG ` 
-SecurityRules $nsgBackendRule
```

Add the network security groups to the subnets:

```
$vnet = Get-AzVirtualNetwork ` 
-ResourceGroupName myRGNetwork ` 
-Name myVNet
$frontendSubnet = $vnet.Subnets[0]
$backendSubnet = $vnet.Subnets[1]
$frontendSubnetConfig = Set-AzVirtualNetworkSubnetConfig ` 
-VirtualNetwork $vnet ` 
-Name myFrontendSubnet ` 
-AddressPrefix $frontendSubnet.AddressPrefix ` 
-NetworkSecurityGroup $nsgFrontend
$backendSubnetConfig = Set-AzVirtualNetworkSubnetConfig ` 
-VirtualNetwork $vnet ` 
-Name myBackendSubnet ` 
-AddressPrefix $backendSubnet.AddressPrefix ` 
-NetworkSecurityGroup $nsgBackend
Set-AzVirtualNetwork -VirtualNetwork $vnet
```

## Create a back-end VM

The easiest way to create the back-end VM for this tutorial is by using a SQL Server image. This tutorial only creates the VM with the database server, but doesn't provide information about accessing the database.

Create *myBackendNic*.

```
$backendNic = New-AzNetworkInterface `  
    -ResourceGroupName myRGNetwork `  
    -Location EastUS `  
    -Name myBackend `  
    -SubnetId $vnet.Subnets[1].Id
```

Set the username and password needed for the administrator account on the VM with Get-Credential:

```
$cred = Get-Credential
```

Create *myBackendVM*.

```
New-AzVM `  
    -Credential $cred `  
    -Name myBackend `  
    -ImageName "MicrosoftSQLServer:SQL2016SP1-WS2016:Enterprise:latest" `  
    -ResourceGroupName myRGNetwork `  
    -Location "EastUS" `  
    -SubnetName MyBackendSubnet `  
    -VirtualNetworkName myVNet
```

The image in this example has SQL Server installed, but it isn't used in this tutorial. It's included to show you how you can configure a VM to handle web traffic and a VM to handle database management.

## Next steps

In this tutorial, you created and secured Azure networks as related to virtual machines.

- Create a virtual network and subnet
- Create a public IP address
- Create a front-end VM
- Secure network traffic
- Create a back-end VM

To learn about protecting your VM disks, see [Backup and disaster recovery for disks](#).

# Azure Resource Graph sample queries for Azure Virtual Machines

9/21/2022 • 18 minutes to read • [Edit Online](#)

This page is a collection of [Azure Resource Graph](#) sample queries for Azure Virtual Machines. For a complete list of Azure Resource Graph samples, see [Resource Graph samples by Category](#) and [Resource Graph samples by Table](#).

## Sample queries

### Count of OS update installation done

Returns a list of status of OS update installation runs done for your machines in last 7 days.

```
PatchAssessmentResources  
| where type !has 'softwarepatches'  
| extend machineName = tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)),  
    rgName = split(id, '/', 4)  
| extend prop = parse_json(properties)  
| extend lTime = todatetime(prop.lastModifiedDateTime), OS = tostring(prop.osType), installedPatchCount =  
    tostring(prop.installedPatchCount), failedPatchCount = tostring(prop.failedPatchCount), pendingPatchCount =  
    tostring(prop.pendingPatchCount), excludedPatchCount = tostring(prop.excludedPatchCount),  
    notSelectedPatchCount = tostring(prop.notSelectedPatchCount)  
| where lTime > ago(7d)  
| project lTime, RunID=name, machineName, rgName, resourceType, OS, installedPatchCount, failedPatchCount,  
    pendingPatchCount, excludedPatchCount, notSelectedPatchCount
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "PatchAssessmentResources | where type !has 'softwarepatches' | extend machineName =  
    tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)), rgName = split(id, '/',  
    4)) | extend prop = parse_json(properties) | extend lTime = todatetime(prop.lastModifiedDateTime), OS =  
    tostring(prop.osType), installedPatchCount = tostring(prop.installedPatchCount), failedPatchCount =  
    tostring(prop.failedPatchCount), pendingPatchCount = tostring(prop.pendingPatchCount), excludedPatchCount =  
    tostring(prop.excludedPatchCount), notSelectedPatchCount = tostring(prop.notSelectedPatchCount) | where  
    lTime > ago(7d) | project lTime, RunID=name, machineName, rgName, resourceType, OS, installedPatchCount,  
    failedPatchCount, pendingPatchCount, excludedPatchCount, notSelectedPatchCount"
```

### Count of virtual machines by availability state and Subscription Id

Returns the count of virtual machines (type `Microsoft.Compute/virtualMachines`) aggregated by their availability state across each of your subscriptions.

```
HealthResources  
| where type =~ 'microsoft.resourcehealth/availabilitystatuses'  
| summarize count() by subscriptionId, AvailabilityState = tostring(properties.availabilityState)
```

- [Azure CLI](#)
- [Azure PowerShell](#)

- [Portal](#)

```
az graph query -q "HealthResources | where type =~ 'microsoft.resourcehealth/availabilitystatuses' | summarize count() by subscriptionId, AvailabilityState = tostring(properties.availabilityState)"
```

## Count of virtual machines by power state

Returns count of virtual machines (type `Microsoft.Compute/virtualMachines`) categorized according to their power state. For more information on power states, please see [Power states overview](#).

```
Resources
| where type == 'microsoft.compute/virtualmachines'
| summarize count() by PowerState = tostring(properties.extended.instanceView.powerState.code)
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type == 'microsoft.compute/virtualmachines' | summarize count() by PowerState = tostring(properties.extended.instanceView.powerState.code)"
```

## Count virtual machines by OS type

Building on the previous query, we're still limiting by Azure resources of type

`Microsoft.Compute/virtualMachines`, but are no longer limiting the number of records returned. Instead, we used `summarize` and `count()` to define how to group and aggregate the values by property, which in this example is `properties.storageProfile.osDisk.osType`. For an example of how this string looks in the full object, see [explore resources - virtual machine discovery](#).

```
Resources
| where type =~ 'Microsoft.Compute/virtualMachines'
| summarize count() by tostring(properties.storageProfile.osDisk.osType)
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type =~ 'Microsoft.Compute/virtualMachines' | summarize count() by tostring(properties.storageProfile.osDisk.osType)"
```

## Count virtual machines by OS type with extend

A different way to write the 'Count virtual machines by OS type' query is to `extend` a property and give it a temporary name for use within the query, in this case `os`. `os` is then used by `summarize` and `count()` as in the referenced example.

```
Resources
| where type =~ 'Microsoft.Compute/virtualMachines'
| extend os = properties.storageProfile.osDisk.osType
| summarize count() by tostring(os)
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type =~ 'Microsoft.Compute/virtualMachines' | extend os = properties.storageProfile.osDisk.osType | summarize count() by tostring(os)"
```

### Get all New alerts from the past 30 days

This query provides a list of all the user's New alerts, from the past 30 days.

```
iotsecurityresources
| where type == 'microsoft.iotsecurity/locations/devicegroups/alerts'
| where todatetime(properties.startTimeUtc) > ago(30d) and properties.status == 'New'
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "iotsecurityresources | where type ==
'microsoft.iotsecurity/locations/devicegroups/alerts' | where todatetime(properties.startTimeUtc) > ago(30d)
and properties.status == 'New'"
```

### Get virtual machine scale set capacity and size

This query looks for virtual machine scale set resources and gets various details including the virtual machine size and the capacity of the scale set. The query uses the `toint()` function to cast the capacity to a number so that it can be sorted. Finally, the columns are renamed into custom named properties.

```
Resources
| where type=~ 'microsoft.compute/virtualmachinescalesets'
| where name contains 'contoso'
| project subscriptionId, name, location, resourceGroup, Capacity = toint(sku.capacity), Tier = sku.name
| order by Capacity desc
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type=~ 'microsoft.compute/virtualmachinescalesets' | where name
contains 'contoso' | project subscriptionId, name, location, resourceGroup, Capacity = toint(sku.capacity),
Tier = sku.name | order by Capacity desc"
```

### List all extensions installed on a virtual machine

First, this query uses `extend` on the virtual machines resource type to get the ID in uppercase (`toupper()`) the ID, get the operating system name and type, and get the virtual machine size. Getting the resource ID in uppercase is a good way to prepare to join to another property. Then, the query uses `join` with `kind as leftouter` to get virtual machine extensions by matching an uppercase `substring` of the extension ID. The portion of the ID before "/extensions/<ExtensionName>" is the same format as the virtual machines ID, so we use this property for the `join`. `summarize` is then used with `make_list` on the name of the virtual machine extension to combine the name of each extension where `id`, `OSName`, `OSType`, and `VMSize` are the same into a

single array property. Lastly, we `order by` lowercase *OSName* with `asc`. By default, `order by` is descending.

```
Resources
| where type == 'microsoft.compute/virtualmachines'
| extend
  JoinID = toupper(id),
  OSName = tostring(properties.osProfile.computerName),
  OSType = tostring(properties.storageProfile.osDisk.osType),
  VMSize = tostring(properties.hardwareProfile.vmSize)
| join kind=leftouter(
  Resources
  | where type == 'microsoft.compute/virtualmachines/extensions'
  | extend
    VMId = toupper(substring(id, 0, indexof(id, '/extensions'))),
    ExtensionName = name
) on $left.JoinID == $right.VMId
| summarize Extensions = make_list(ExtensionName) by id, OSName, OSType, VMSize
| order by tolower(OSName) asc
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type == 'microsoft.compute/virtualmachines' | extend JoinID =
toupper(id), OSName = tostring(properties.osProfile.computerName), OSType =
tostring(properties.storageProfile.osDisk.osType), VMSize = tostring(properties.hardwareProfile.vmSize) |
join kind=leftouter( Resources | where type == 'microsoft.compute/virtualmachines/extensions' | extend VMId =
toupper(substring(id, 0, indexof(id, '/extensions'))), ExtensionName = name ) on \$left.JoinID ==
\$right.VMId | summarize Extensions = make_list(ExtensionName) by id, OSName, OSType, VMSize | order by
tolower(OSName) asc"
```

## List available OS updates for all your machines grouped by update category

Returns a list of pending OS for your machines.

```
PatchAssessmentResources
| where type !has 'softwarepatches'
| extend prop = parse_json(properties)
| extend lastTime = properties.lastModifiedDateTime
| extend updateRollupCount = prop.availablePatchCountByClassification.updateRollup, featurePackCount =
prop.availablePatchCountByClassification.featurePack, servicePackCount =
prop.availablePatchCountByClassification.servicePack, definitionCount =
prop.availablePatchCountByClassification.definition, securityCount =
prop.availablePatchCountByClassification.security, criticalCount =
prop.availablePatchCountByClassification.critical, updatesCount =
prop.availablePatchCountByClassification.updates, toolsCount =
prop.availablePatchCountByClassification.tools, otherCount = prop.availablePatchCountByClassification.other,
OS = prop.osType
| project lastTime, id, OS, updateRollupCount, featurePackCount, servicePackCount, definitionCount,
securityCount, criticalCount, updatesCount, toolsCount, otherCount
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```

az graph query -q "PatchAssessmentResources | where type !has 'softwarepatches' | extend prop =
parse_json(properties) | extend lastTime = properties.lastModifiedDateTime | extend updateRollupCount =
prop.availablePatchCountByClassification.updateRollup, featurePackCount =
prop.availablePatchCountByClassification.featurePack, servicePackCount =
prop.availablePatchCountByClassification.servicePack, definitionCount =
prop.availablePatchCountByClassification.definition, securityCount =
prop.availablePatchCountByClassification.security, criticalCount =
prop.availablePatchCountByClassification.critical, updatesCount =
prop.availablePatchCountByClassification.updates, toolsCount =
prop.availablePatchCountByClassification.tools, otherCount = prop.availablePatchCountByClassification.other,
OS = prop.osType | project lastTime, id, OS, updateRollupCount, featurePackCount, servicePackCount,
definitionCount, securityCount, criticalCount, updatesCount, toolsCount, otherCount"

```

## List of Linux OS update installation done

Returns a list of status of Linux Server - OS update installation runs done for your machines in last 7 days.

```

PatchAssessmentResources
| where type has 'softwarepatches' and properties has 'version'
| extend machineName = tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)),
tostring(rgName = split(id, '/', 4)), tostring(RunID = split(id, '/', 10))
| extend prop = parse_json(properties)
| extend lTime = todatetime(prop.lastModifiedDateTime), patchName = tostring(prop.patchName), version =
tostring(prop.version), installationState = tostring(prop.installationState), classifications =
tostring(prop.classifications)
| where lTime > ago(7d)
| project lTime, RunID, machineName, rgName, resourceType, patchName, version, classifications,
installationState
| sort by RunID

```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```

az graph query -q "PatchAssessmentResources | where type has 'softwarepatches' and properties has 'version'
| extend machineName = tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)),
tostring(rgName = split(id, '/', 4)), tostring(RunID = split(id, '/', 10)) | extend prop =
parse_json(properties) | extend lTime = todatetime(prop.lastModifiedDateTime), patchName =
tostring(prop.patchName), version = tostring(prop.version), installationState =
tostring(prop.installationState), classifications = tostring(prop.classifications) | where lTime > ago(7d) |
project lTime, RunID, machineName, rgName, resourceType, patchName, version, classifications,
installationState | sort by RunID"

```

## List of virtual machines and associated availability states by Resource Ids

Returns the latest list of virtual machines (type `Microsoft.Compute/virtualMachines`) aggregated by availability state. The query also provides the associated Resource Id based on `properties.targetResourceId`, for easy debugging and mitigation. Availability states can be one of four values: Available, Unavailable, Degraded and Unknown. For more details on what each of the availability states mean, please see [Azure Resource Health overview](#).

```

HealthResources
| where type =~ 'microsoft.resourcehealth/availabilitystatuses'
| summarize by ResourceId = tolower(tostring(properties.targetResourceId)), AvailabilityState =
tostring(properties.availabilityState)

```

- [Azure CLI](#)
- [Azure PowerShell](#)

- [Portal](#)

```
az graph query -q "HealthResources | where type =~ 'microsoft.resourcehealth/availabilitystatuses' | summarize by ResourceId = tolower(tostring(properties.targetResourceId)), AvailabilityState = tostring(properties.availabilityState)"
```

### List of virtual machines by availability state and power state with Resource Ids and resource Groups

Returns list of virtual machines (type `Microsoft.Compute/virtualMachines`) aggregated on their power state and availability state to provide a cohesive state of health for your virtual machines. The query also provides details on the resource group and resource Id associated with each entry for detailed visibility into your resources.

```
Resources
| where type =~ 'microsoft.compute/virtualmachines'
| project resourceGroup, Id = tolower(id), PowerState = tostring(
properties.extended.instanceView.powerState.code)
| join kind=leftouter (
HealthResources
| where type =~ 'microsoft.resourcehealth/availabilitystatuses'
| where tostring(properties.targetResourceType) =~ 'microsoft.compute/virtualmachines'
| project targetResourceId = tolower(tostring(properties.targetResourceId)), AvailabilityState =
tostring(properties.availabilityState))
on $left.Id == $right.targetResourceId
| project-away targetResourceId
| where PowerState != 'PowerState/deallocated'
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type =~ 'microsoft.compute/virtualmachines' | project resourceGroup, Id = tolower(id), PowerState = tostring( properties.extended.instanceView.powerState.code) | join kind=leftouter ( HealthResources | where type =~ 'microsoft.resourcehealth/availabilitystatuses' | where tostring(properties.targetResourceType) =~ 'microsoft.compute/virtualmachines' | project targetResourceId = tolower(tostring(properties.targetResourceId)), AvailabilityState = tostring(properties.availabilityState)) on \$left.Id == \$right.targetResourceId | project-away targetResourceId | where PowerState != 'PowerState/deallocated'"
```

### List of virtual machines that are not Available by Resource Ids

Returns the latest list of virtual machines (type `Microsoft.Compute/virtualMachines`) aggregated by their availability state. The populated list only highlights virtual machines whose availability state is not "Available" to ensure you are aware of all the concerning states your virtual machines are in. When all your virtual machines are Available, you can expect to receive no results.

```
HealthResources
| where type =~ 'microsoft.resourcehealth/availabilitystatuses'
| where tostring(properties.availabilityState) != 'Available'
| summarize by ResourceId = tolower(tostring(properties.targetResourceId)), AvailabilityState = tostring(properties.availabilityState)
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "HealthResources | where type =~ 'microsoft.resourcehealth/availabilitystatuses' | where
tostring(properties.availabilityState) != 'Available' | summarize by ResourceId =
tolower(tostring(properties.targetResourceId)), AvailabilityState = tostring(properties.availabilityState)"
```

## List of Windows Server OS update installation done

Returns a list of status of Windows Server - OS update installation runs done for your machines in last 7 days.

```
PatchAssessmentResources
| where type has 'softwarepatches' and properties !has 'version'
| extend machineName = tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)),
tostring(rgName = split(id, '/', 4)), tostring(RunID = split(id, '/', 10))
| extend prop = parse_json(properties)
| extend lTime = todatetime(prop.lastModifiedDateTime), patchName = tostring(prop.patchName), kbId =
tostring(prop.kbId), installationState = tostring(prop.installationState), classifications =
tostring(prop.classifications)
| where lTime > ago(7d)
| project lTime, RunID, machineName, rgName, resourceType, patchName, kbId, classifications,
installationState
| sort by RunID
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "PatchAssessmentResources | where type has 'softwarepatches' and properties !has 'version'
| extend machineName = tostring(split(id, '/', 8)), resourceType = tostring(split(type, '/', 0)),
tostring(rgName = split(id, '/', 4)), tostring(RunID = split(id, '/', 10)) | extend prop =
parse_json(properties) | extend lTime = todatetime(prop.lastModifiedDateTime), patchName =
tostring(prop.patchName), kbId = tostring(prop.kbId), installationState = tostring(prop.installationState),
classifications = tostring(prop.classifications) | where lTime > ago(7d) | project lTime, RunID,
machineName, rgName, resourceType, patchName, kbId, classifications, installationState | sort by RunID"
```

## List virtual machines with their network interface and public IP

This query uses two **leftouter join** commands to bring together virtual machines created with the Resource Manager deployment model, their related network interfaces, and any public IP address related to those network interfaces.

```

Resources
| where type =~ 'microsoft.compute/virtualmachines'
| extend nics=array_length(properties.networkProfile.networkInterfaces)
| mv-expand nic=properties.networkProfile.networkInterfaces
| where nics == 1 or nic.properties.primary =~ 'true' or isempty(nic)
| project vmId = id, vmName = name, vmSize=toString(properties.hardwareProfile.vmSize), nicId =
toString(nic.id)
| join kind=leftouter (
    Resources
    | where type =~ 'microsoft.network/networkinterfaces'
    | extend ipConfigsCount=array_length(properties.ipConfigurations)
    | mv-expand ipconfig=properties.ipConfigurations
    | where ipConfigsCount == 1 or ipconfig.properties.primary =~ 'true'
    | project nicId = id, publicIpId = toString(ipconfig.properties.publicIPAddress.id))
on nicId
| project-away nicId1
| summarize by vmId, vmName, vmSize, nicId, publicIpId
| join kind=leftouter (
    Resources
    | where type =~ 'microsoft.network/publicipaddresses'
    | project publicIpId = id, publicIpAddress = properties.ipAddress)
on publicIpId
| project-away publicIpId1

```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```

az graph query -q "Resources | where type =~ 'microsoft.compute/virtualmachines' | extend
nics=array_length(properties.networkProfile.networkInterfaces) | mv-expand
nic=properties.networkProfile.networkInterfaces | where nics == 1 or nic.properties.primary =~ 'true' or
isempty(nic) | project vmId = id, vmName = name, vmSize=toString(properties.hardwareProfile.vmSize), nicId =
toString(nic.id) | join kind=leftouter ( Resources | where type =~ 'microsoft.network/networkinterfaces' |
extend ipConfigsCount=array_length(properties.ipConfigurations) | mv-expand
ipconfig=properties.ipConfigurations | where ipConfigsCount == 1 or ipconfig.properties.primary =~ 'true' |
project nicId = id, publicIpId = toString(ipconfig.properties.publicIPAddress.id)) on nicId | project-away
nicId1 | summarize by vmId, vmName, vmSize, nicId, publicIpId | join kind=leftouter ( Resources | where type
=~ 'microsoft.network/publicipaddresses' | project publicIpId = id, publicIpAddress = properties.ipAddress)
on publicIpId | project-away publicIpId1"

```

### Show all virtual machines ordered by name in descending order

To list only virtual machines (which are type `Microsoft.Compute/virtualMachines`), we can match the property `type` in the results. Similar to the previous query, `desc` changes the `order by` to be descending. The `=~` in the type match tells Resource Graph to be case insensitive.

```

Resources
| project name, location, type
| where type =~ 'Microsoft.Compute/virtualMachines'
| order by name desc

```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```

az graph query -q "Resources | project name, location, type | where type =~
'Microsoft.Compute/virtualMachines' | order by name desc"

```

## Show first five virtual machines by name and their OS type

This query uses `top` to only retrieve five matching records that are ordered by name. The type of the Azure resource is `Microsoft.Compute/virtualMachines`. `project` tells Azure Resource Graph which properties to include.

```
Resources
| where type =~ 'Microsoft.Compute/virtualMachines'
| project name, properties.storageProfile.osDisk.osType
| top 5 by name desc
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type =~ 'Microsoft.Compute/virtualMachines' | project name,
properties.storageProfile.osDisk.osType | top 5 by name desc"
```

## Summarize virtual machine by the power states extended property

This query uses the [extended properties](#) on virtual machines to summarize by power states.

```
Resources
| where type == 'microsoft.compute/virtualmachines'
| summarize count() by tostring(properties.extended.instanceView.powerState.code)
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type == 'microsoft.compute/virtualmachines' | summarize count() by
tostring(properties.extended.instanceView.powerState.code)"
```

## Virtual machines matched by regex

This query looks for virtual machines that match a [regular expression](#) (known as *regex*). The `matches regex @` allows us to define the regex to match, which is `^Contoso(.*)[0-9]+$`. That regex definition is explained as:

- `^` - Match must start at the beginning of the string.
- `Contoso` - The case-sensitive string.
- `(.*)` - A subexpression match:
  - `.` - Matches any single character (except a new line).
  - `*` - Matches previous element zero or more times.
- `[0-9]` - Character group match for numbers 0 through 9.
- `+` - Matches previous element one or more times.
- `$` - Match of the previous element must occur at the end of the string.

After matching by name, the query projects the name and orders by name ascending.

```
Resources
| where type =~ 'microsoft.compute/virtualmachines' and name matches regex @'^Contoso(.*?[0-9]+$'
| project name
| order by name asc
```

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)

```
az graph query -q "Resources | where type =~ 'microsoft.compute/virtualmachines' and name matches regex
@'^Contoso(.*?[0-9]+$' | project name | order by name asc"
```

## Next steps

- Learn more about the [query language](#).
- Learn more about how to [explore resources](#).
- See samples of [Starter language queries](#).
- See samples of [Advanced language queries](#).

# Red Hat workloads on Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Red Hat workloads are supported through a variety of offerings on Azure. Red Hat Enterprise Linux (RHEL) images are at the core of RHEL workloads, as is the Red Hat Update Infrastructure (RHUI).

## Red Hat Enterprise Linux images

Azure offers a wide offering of RHEL images on Azure. These images are made available through two different licensing models: pay-as-you-go and bring-your-own-subscription (BYOS). New RHEL images on Azure are published when new RHEL versions are released and updated throughout their lifecycles, as necessary.

### Pay-as-you-go images

Azure offers a variety of RHEL pay-as-you-go images. These images come properly entitled for RHEL and are attached to a source of updates (Red Hat Update Infrastructure). These images charge a premium fee for the RHEL entitlement and updates. RHEL pay-as-you-go image variants include:

- RHEL
- RHEL for SAP
- RHEL for SAP with High Availability (HA) and Update Services

You might want to use the pay-as-you-go images if you don't want to worry about paying separately for the appropriate number of subscriptions.

### Red Hat Gold Images

Azure also offers Red Hat Gold Images (`rhel1-byos`). These images might be useful to customers who have existing Red Hat subscriptions and want to use them in Azure. You're required to enable your existing Red Hat subscriptions for Red Hat Cloud Access before you can use them in Azure. Access to these images is granted automatically when your Red Hat subscriptions are enabled for Cloud Access and meet the eligibility requirements. Using these images allows a customer to avoid double billing that might be incurred from using the pay-as-you-go images.

- Learn how to [enable your Red Hat subscriptions for Cloud Access with Azure](#).
- Learn how to [locate Red Hat Gold Images in the Azure portal, the Azure CLI, or PowerShell cmdlet](#).

#### NOTE

Double billing is incurred when a user pays twice for RHEL subscriptions. This scenario usually happens when a customer uses Red Hat Subscription-Manager to attach an entitlement on a RHEL pay-as-you-go VM. For example, a customer who uses Subscription-Manager to attach an entitlement for SAP packages on a RHEL pay-as-you-go image is indirectly double billed because they pay twice for RHEL. They pay once through the pay-as-you-go premium fee and once through their SAP subscription. This scenario doesn't happen to BYOS image users.

## Generation 2 images

Generation 2 virtual machines (VMs) provide some newer features compared to Generation 1 VMs. For more information, see the [Generation 2 documentation](#). The key difference from a RHEL image perspective is that Generation 2 VMs use a UEFI instead of BIOS firmware interface. They also use a GUID Partition Table (GPT) instead of a master boot record (MBR) on boot time. Use of a GPT allows for, among other things, OS disk sizes

larger than 2 TB. In addition, the [Mv2 series VMs](#) run only on Generation 2 images.

RHEL Generation 2 images are available in the Azure Marketplace. Look for "gen2" in the image SKU in the list of all images that appears when you use the Azure CLI. Go to the **Advanced** tab in the VM deploy process to deploy a Generation 2 VM.

## Red Hat Update Infrastructure

Azure provides Red Hat Update Infrastructure only for pay-as-you-go RHEL VMs. RHUI is effectively a mirror of the Red Hat CDNs but is only accessible to the Azure pay-as-you-go RHEL VMs. You have access to the appropriate packages depending on which RHEL image you've deployed. For example, a RHEL for SAP image has access to the SAP packages in addition to base RHEL packages.

### RHUI update behavior

RHEL images connected to RHUI update by default to the latest minor version of RHEL when a `yum update` is run. This behavior means that a RHEL 7.4 VM might get upgraded to RHEL 7.7 if a `yum update` operation is run on it. This behavior is by design for RHUI. To mitigate this upgrade behavior, switch from regular RHEL repositories to [Extended Update Support repositories](#).

## Red Hat Middleware

Microsoft and Azure have partnered to develop a variety of solutions for running Red Hat Middleware on Azure. Learn more about JBoss EAP on Azure Virtual Machines and Azure App service at [Red Hat JBoss EAP on Azure](#).

## Next steps

- Learn more about [RHEL images on Azure](#).
- Learn more about [Red Hat Update Infrastructure](#).
- Learn more about the [Red Hat Gold Image \( rhel-byos \) offer](#).



# OpenShift in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

OpenShift is an open and extensible container application platform that brings Docker and Kubernetes to the enterprise.

OpenShift includes Kubernetes for container orchestration and management. It adds developer-centric and operations-centric tools that enable:

- Rapid application development.
- Easy deployment and scaling.
- Long-term lifecycle maintenance for teams and applications.

There are multiple versions of OpenShift available. Of these versions, only two are available today for customers to deploy in Azure: OpenShift Container Platform and OKD (formerly OpenShift Origin).

## Azure Red Hat OpenShift

Microsoft Azure Red Hat OpenShift is a fully managed offering of OpenShift running in Azure. This service is jointly managed and supported by Microsoft and Red Hat. For more details, see the [Azure Red Hat OpenShift Service](#) documentation.

## OpenShift Container Platform

Container Platform is an enterprise-ready [commercial version](#) from and supported by Red Hat. With this version, customers purchase the necessary entitlements for OpenShift Container Platform and are responsible for installation and management of the entire infrastructure.

Because customers "own" the entire platform, they can install it in their on-premises datacenter, or in a public cloud (such as Azure).

## OKD

OKD is an [open-source](#) upstream project of OpenShift that's community supported. OKD can be installed on CentOS or Red Hat Enterprise Linux (RHEL).

## Next steps

- [Configure common prerequisites for OpenShift in Azure](#)
- [Deploy OpenShift Container Platform in Azure](#)
- [Deploy OpenShift Container Platform Self-Managed Marketplace Offer](#)
- [Deploy OpenShift in Azure Stack](#)
- [Post-deployment tasks](#)
- [Troubleshoot OpenShift deployment](#)

# Deploy OpenShift Container Platform 4.x in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Deployment of OpenShift Container Platform (OCP) 4.2 is now supported in Azure via the Installer-Provisioned Infrastructure (IPI) model. The landing page for trying OpenShift 4 is [try.openshift.com](https://try.openshift.com). To install OCP 4.2 in Azure, visit the [Red Hat OpenShift Cluster Manager](#) page. Red Hat credentials are required to access this site.

## Notes

- An Azure Active Directory (AAD) Service Principal (SP) is required to install and run OCP 4.x in Azure
  - The SP must be granted the API permission of **Application.ReadWrite.OwnedBy** for Azure Active Directory Graph
  - An AAD Tenant Administrator must grant Admin Consent for this API permission to take effect
  - The SP must be granted **Contributor** and **User Access Administrator** roles to the subscription
- The installation model for OCP 4.x is different than 3.x and there are no Azure Resource Manager templates available for deploying OCP 4.x in Azure
- If issues are encountered during the installation process, contact the appropriate company (Microsoft or Red Hat)

ISSUE DESCRIPTION	CONTACT POINT
Azure specific issues (AAD, SP, Azure Subscription, etc.)	Microsoft
OpenShift-specific issues (Installation failures / errors, Red Hat subscription, etc.)	Red Hat

## Next steps

- [Getting started with OpenShift Container Platform](#)

# Common prerequisites for deploying OpenShift Container Platform 3.11 in Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article describes common prerequisites for deploying OpenShift Container Platform or OKD in Azure.

The installation of OpenShift uses Ansible playbooks. Ansible uses Secure Shell (SSH) to connect to all cluster hosts to complete installation steps.

When ansible makes the SSH connection to the remote hosts, it can't enter a password. For this reason, the private key can't have a password (passphrase) associated with it or deployment fails.

Because the virtual machines (VMs) deploy via Azure Resource Manager templates, the same public key is used for access to all VMs. The corresponding private key must be on the VM that executes all the playbooks as well. To perform this action securely, an Azure key vault is used to pass the private key into the VM.

If there's a need for persistent storage for containers, then persistent volumes are required. OpenShift supports Azure virtual hard disks (VHDs) for persistent volumes, but Azure must first be configured as the cloud provider.

In this model, OpenShift:

- Creates a VHD object in an Azure storage account or a managed disk.
- Mounts the VHD to a VM and formats the volume.
- Mounts the volume to the pod.

For this configuration to work, OpenShift needs permissions to perform these tasks in Azure. A service principal is used for this purpose. The service principal is a security account in Azure Active Directory that is granted permissions to resources.

The service principal needs to have access to the storage accounts and VMs that make up the cluster. If all OpenShift cluster resources deploy to a single resource group, the service principal can be granted permissions to that resource group.

This guide describes how to create the artifacts associated with the prerequisites.

- Create a key vault to manage SSH keys for the OpenShift cluster.
- Create a service principal for use by the Azure Cloud Provider.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Sign in to Azure

Sign in to your Azure subscription with the [az login](#) command and follow the on-screen directions, or click Try it to use Cloud Shell.

```
az login
```

## Create a resource group

Create a resource group with the [az group create](#) command. An Azure resource group is a logical container into

which Azure resources are deployed and managed. You should use a dedicated resource group to host the key vault. This group is separate from the resource group into which the OpenShift cluster resources deploy.

The following example creates a resource group named *keyvaultrg* in the *eastus* location:

```
az group create --name keyvaultrg --location eastus
```

## Create a key vault

Create a key vault to store the SSH keys for the cluster with the [az keyvault create](#) command. The key vault name must be globally unique and must be enabled for template deployment or the deployment will fail with "KeyVaultParameterReferenceSecretRetrieveFailed" error.

The following example creates a key vault named *keyvault* in the *keyvaultrg* resource group:

```
az keyvault create --resource-group keyvaultrg --name keyvault \
    --enabled-for-template-deployment true \
    --location eastus
```

## Create an SSH key

An SSH key is needed to secure access to the OpenShift cluster. Create an SSH key pair by using the [ssh-keygen](#) command (on Linux or macOS):

```
ssh-keygen -f ~/.ssh/openshift_rsa -t rsa -N ''
```

### NOTE

Your SSH key pair can't have a password / passphrase.

For more information on SSH keys on Windows, see [How to create SSH keys on Windows](#). Be sure to export the private key in OpenSSH format.

## Store the SSH private key in Azure Key Vault

The OpenShift deployment uses the SSH key you created to secure access to the OpenShift master. To enable the deployment to securely retrieve the SSH key, store the key in Key Vault by using the following command:

```
az keyvault secret set --vault-name keyvault --name keysecret --file ~/.ssh/openshift_rsa
```

## Create a service principal

OpenShift communicates with Azure by using a username and password or a service principal. An Azure service principal is a security identity that you can use with apps, services, and automation tools like OpenShift. You control and define the permissions as to which operations the service principal can perform in Azure. It's best to scope the permissions of the service principal to specific resource groups rather than the entire subscription.

Create a service principal with [az ad sp create-for-rbac](#) and output the credentials that OpenShift needs.

The following example creates a service principal and assigns it contributor permissions to a resource group named *openshifttrg*.

First, create the resource group named *openshiftrg*.

```
az group create -l eastus -n openshiftrg
```

Create service principal:

```
az group show --name openshiftrg --query id
```

Save the output of the command and use in place of \$scope in next command

```
az ad sp create-for-rbac --name openshiftsp \
--role Contributor --scopes $scope \
```

Take note of the appId property and password returned from the command:

```
{
  "appId": "11111111-abcd-1234-efgh-111111111111",
  "displayName": "openshiftsp",
  "name": "http://openshiftsp",
  "password": {Strong Password},
  "tenant": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX"
}
```

#### WARNING

Be sure to write down the secure password as it will not be possible to retrieve this password again.

For more information on service principals, see [Create an Azure service principal with Azure CLI](#).

## Prerequisites applicable only to Resource Manager template

Secrets will need to be created for the SSH private key (**sshPrivateKey**), Azure AD client secret (**aadClientSecret**), OpenShift admin password (**openshiftPassword**), and Red Hat Subscription Manager password or activation key (**rhsmPasswordOrActivationKey**). Additionally, if custom TLS/SSL certificates are used, then six additional secrets will need to be created - **routingcafile**, **routingcertfile**, **routingkeyfile**, **mastercafile**, **mastercertfile**, and **masterkeyfile**. These parameters will be explained in more detail.

The template references specific secret names so you **must** use the bolded names listed above (case sensitive).

### Custom Certificates

By default, the template will deploy an OpenShift cluster using self-signed certificates for the OpenShift web console and the routing domain. If you want to use custom TLS/SSL certificates, set 'routingCertType' to 'custom' and 'masterCertType' to 'custom'. You'll need the CA, Cert, and Key files in .pem format for the certificates. It is possible to use custom certificates for one but not the other.

You'll need to store these files in Key Vault secrets. Use the same Key Vault as the one used for the private key. Rather than require 6 additional inputs for the secret names, the template is hard-coded to use specific secret names for each of the TLS/SSL certificate files. Store the certificate data using the information from the following table.

SECRET NAME	CERTIFICATE FILE
mastercafile	master CA file
mastercertfile	master CERT file
masterkeyfile	master Key file
routingcafile	routing CA file
routingcertfile	routing CERT file
routingkeyfile	routing Key file

Create the secrets using the Azure CLI. Below is an example.

```
az keyvault secret set --vault-name KeyVaultName -n mastercafile --file ~/certificates/masterca.pem
```

## Next steps

This article covered the following topics:

- Create a key vault to manage SSH keys for the OpenShift cluster.
- Create a service principal for use by the Azure Cloud Solution Provider.

Next, deploy an OpenShift cluster:

- [Deploy OpenShift Container Platform](#)
- [Deploy OpenShift Container Platform Self-Managed Marketplace Offer](#)

# Deploy OpenShift Container Platform 3.11 in Azure

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You can use one of several methods to deploy OpenShift Container Platform 3.11 in Azure:

- You can manually deploy the necessary Azure infrastructure components and then follow the [OpenShift Container Platform documentation](#).
- You can also use an existing [Resource Manager template](#) that simplifies the deployment of the OpenShift Container Platform cluster.
- Another option is to use the Azure Marketplace offer.

For all options, a Red Hat subscription is required. During the deployment, the Red Hat Enterprise Linux instance is registered to the Red Hat subscription and attached to the Pool ID that contains the entitlements for OpenShift Container Platform. Make sure you have a valid Red Hat Subscription Manager (RHSM) username, password, and Pool ID. You can use an Activation Key, Org ID, and Pool ID. You can verify this information by signing in to <https://access.redhat.com>.

## Deploy using the OpenShift Container Platform Resource Manager 3.11 template

### Private Clusters

Deploying private OpenShift clusters requires more than just not having a public IP associated to the master load balancer (web console) or to the infra load balancer (router). A private cluster generally uses a custom DNS server (not the default Azure DNS), a custom domain name (such as contoso.com), and pre-defined virtual network(s). For private clusters, you need to configure your virtual network with all the appropriate subnets and DNS server settings in advance. Then use `existingMasterSubnetReference`, `existingInfraSubnetReference`, `existingCnsSubnetReference`, and `existingNodeSubnetReference` to specify the existing subnet for use by the cluster.

If private master is selected (`masterClusterType=private`), a static private IP needs to be specified for `masterPrivateClusterIp`. This IP will be assigned to the front end of the master load balancer. The IP must be within the CIDR for the master subnet and not in use. `masterClusterDnsType` must be set to "custom" and the master DNS name must be provided for `masterClusterDns`. The DNS name must map to the static Private IP and will be used to access the console on the master nodes.

If private router is selected (`routerClusterType=private`), a static private IP needs to be specified for `routerPrivateClusterIp`. This IP will be assigned to the front end of the infra load balancer. The IP must be within the CIDR for the infra subnet and not in use. `routingSubDomainType` must be set to "custom" and the wildcard DNS name for routing must be provided for `routingSubDomain`.

If private masters and private router are selected, the custom domain name must also be entered for `domainName`

After successful deployment, the Bastion Node is the only node with a public IP that you can ssh into. Even if the master nodes are configured for public access, they aren't exposed for ssh access.

To deploy using the Resource Manager template, you use a parameters file to supply the input parameters. To further customize the deployment, fork the GitHub repo and change the appropriate items.

Some common customization options include, but aren't limited to:

- Bastion VM size (variable in azuredeploy.json)
- Naming conventions (variables in azuredeploy.json)
- OpenShift cluster specifics, modified via hosts file (deployOpenShift.sh)

## Configure the parameters file

The [OpenShift Container Platform template](#) has multiple branches available for different versions of OpenShift Container Platform. Based on your needs, you can deploy directly from the repo or you can fork the repo and make custom changes to the templates or scripts before deploying.

Use the `appId` value from the service principal you created earlier for the `aadClientId` parameter.

The following example shows a parameters file named `azuredeploy.parameters.json` with all the required inputs.

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "_artifactsLocation": {
      "value": "https://raw.githubusercontent.com/Microsoft/openshift-container-platform/master"
    },
    "location": {
      "value": "eastus"
    },
    "masterVmSize": {
      "value": "Standard_E2s_v3"
    },
    "infraVmSize": {
      "value": "Standard_D4s_v3"
    },
    "nodeVmSize": {
      "value": "Standard_D4s_v3"
    },
    "cnsVmSize": {
      "value": "Standard_E4s_v3"
    },
    "osImageType": {
      "value": "defaultgallery"
    },
    "marketplaceOsImage": {
      "value": {
        "publisher": "RedHat",
        "offer": "RHEL",
        "sku": "7-RAW",
        "version": "latest"
      }
    },
    "storageKind": {
      "value": "changeme"
    },
    "openshiftClusterPrefix": {
      "value": "changeme"
    },
    "minorVersion": {
      "value": "69"
    },
    "masterInstanceCount": {
      "value": 3
    },
    "infraInstanceCount": {
      "value": 3
    },
    "nodeInstanceCount": {
      "value": 3
    },
    "cnsInstanceCount": {
      "value": 2
    }
  }
}
```

```
"value": 3
},
"osDiskSize": {
    "value": 64
},
"dataDiskSize": {
    "value": 64
},
"cnsGlusterDiskSize": {
    "value": 128
},
"adminUsername": {
    "value": "changeme"
},
"enableMetrics": {
    "value": "false"
},
"enableLogging": {
    "value": "false"
},
"enableCNS": {
    "value": "false"
},
"rhsmUsernameOrOrgId": {
    "value": "changeme"
},
"rhsmPoolId": {
    "value": "changeme"
},
"rhsmBrokerPoolId": {
    "value": "changeme"
},
"sshPublicKey": {
    "value": "GEN-SSH-PUB-KEY"
},
"keyVaultSubscriptionId": {
    "value": "255a325e-8276-4ada-af8f-33af5658eb34"
},
"keyVaultResourceGroup": {
    "value": "changeme"
},
"keyVaultName": {
    "value": "changeme"
},
"enableAzure": {
    "value": "true"
},
"aadClientId": {
    "value": "changeme"
},
"domainName": {
    "value": "contoso.com"
},
"masterClusterDnsType": {
    "value": "default"
},
"masterClusterDns": {
    "value": "console.contoso.com"
},
"routingSubDomainType": {
    "value": "nipo"
},
"routingSubDomain": {
    "value": "apps.contoso.com"
},
"virtualNetworkNewOrExisting": {
    "value": "new"
},
"virtualNetworkName": {
```

```

        "value": "changeme"
    },
    "addressPrefixes": {
        "value": "10.0.0.0/14"
    },
    "masterSubnetName": {
        "value": "changeme"
    },
    "masterSubnetPrefix": {
        "value": "10.1.0.0/16"
    },
    "infraSubnetName": {
        "value": "changeme"
    },
    "infraSubnetPrefix": {
        "value": "10.2.0.0/16"
    },
    "nodeSubnetName": {
        "value": "changeme"
    },
    "nodeSubnetPrefix": {
        "value": "10.3.0.0/16"
    },
    "existingMasterSubnetReference": {
        "value": "/subscriptions/abc686f6-963b-4e64-bff4-
99dc369ab1cd/resourceGroups/vnetresourcegroup/providers/Microsoft.Network/virtualNetworks/openshiftvnet/subn
ets/mastersubnet"
    },
    "existingInfraSubnetReference": {
        "value": "/subscriptions/abc686f6-963b-4e64-bff4-
99dc369ab1cd/resourceGroups/vnetresourcegroup/providers/Microsoft.Network/virtualNetworks/openshiftvnet/subn
ets/infrasubnet"
    },
    "existingCnsSubnetReference": {
        "value": "/subscriptions/abc686f6-963b-4e64-bff4-
99dc369ab1cd/resourceGroups/vnetresourcegroup/providers/Microsoft.Network/virtualNetworks/openshiftvnet/subn
ets/cnssubnet"
    },
    "existingNodeSubnetReference": {
        "value": "/subscriptions/abc686f6-963b-4e64-bff4-
99dc369ab1cd/resourceGroups/vnetresourcegroup/providers/Microsoft.Network/virtualNetworks/openshiftvnet/subn
ets/nodesubnet"
    },
    "masterClusterType": {
        "value": "public"
    },
    "masterPrivateClusterIp": {
        "value": "10.1.0.200"
    },
    "routerClusterType": {
        "value": "public"
    },
    "routerPrivateClusterIp": {
        "value": "10.2.0.200"
    },
    "routingCertType": {
        "value": "selfsigned"
    },
    "masterCertType": {
        "value": "selfsigned"
    }
}
}

```

Replace the parameters with your specific information.

Different releases may have different parameters so verify the necessary parameters for the branch you use.

## azuredeploy.Parameters.json file explained

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>_artifactsLocation</code>	URL for artifacts (json, scripts, etc.)		<a href="https://raw.githubusercontent.com/Microsoft/openshift-container-platform/master">https://raw.githubusercontent.com/Microsoft/openshift-container-platform/master</a>
<code>location</code>	Azure region to deploy resources to		
<code>masterVmSize</code>	Size of the Master VM. Select from one of the allowed VM sizes listed in the azuredeploy.json file		Standard_E2s_v3
<code>infraVmSize</code>	Size of the Infra VM. Select from one of the allowed VM sizes listed in the azuredeploy.json file		Standard_D4s_v3
<code>nodeVmSize</code>	Size of the App Node VM. Select from one of the allowed VM sizes listed in the azuredeploy.json file		Standard_D4s_v3
<code>cnsVmSize</code>	Size of the Container Native Storage (CNS) Node VM. Select from one of the allowed VM sizes listed in the azuredeploy.json file		Standard_E4s_v3
<code>osImageType</code>	The RHEL image to use. defaultgallery: On-Demand; marketplace: third-party image	defaultgallery marketplace	defaultgallery
<code>marketplaceOsImage</code>	If <code>osImageType</code> is marketplace, then enter the appropriate values for 'publisher', 'offer', 'sku', 'version' of the marketplace offer. This parameter is an object type		
<code>storageKind</code>	The type of storage to be used	managed unmanaged	managed
<code>openshiftClusterPrefix</code>	Cluster Prefix used to configure hostnames for all nodes. Between 1 and 20 characters		mycluster
<code>minoVersion</code>	The minor version of OpenShift Container Platform 3.11 to deploy		69
<code>masterInstanceCount</code>	Number of Masters nodes to deploy	1, 3, 5	3

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>infraInstanceCount</code>	Number of infra nodes to deploy	1, 2, 3	3
<code>nodeInstanceCount</code>	Number of Nodes to deploy	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30	2
<code>cnsInstanceCount</code>	Number of CNS nodes to deploy	3, 4	3
<code>osDiskSize</code>	Size of OS disk for the VM (in GB)	64, 128, 256, 512, 1024, 2048	64
<code>dataDiskSize</code>	Size of data disk to attach to nodes for Docker volume (in GB)	32, 64, 128, 256, 512, 1024, 2048	64
<code>cnsGlusterDiskSize</code>	Size of data disk to attach to CNS nodes for use by glusterfs (in GB)	32, 64, 128, 256, 512, 1024, 2048	128
<code>adminUsername</code>	Admin username for both OS (VM) login and initial OpenShift user		ocpadmin
<code>enableMetrics</code>	Enable Metrics. Metrics require more resources so select proper size for Infra VM	true false	false
<code>enableLogging</code>	Enable Logging. elasticsearch pod requires 8 GB RAM so select proper size for Infra VM	true false	false
<code>enableCNS</code>	Enable Container Native Storage	true false	false
<code>rhsmUsernameOrOrgId</code>	Red Hat Subscription Manager Username or Organization ID		
<code>rhsmPoolId</code>	The Red Hat Subscription Manager Pool ID that contains your OpenShift entitlements for compute nodes		

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>rhsmBrokerPoolId</code>	The Red Hat Subscription Manager Pool ID that contains your OpenShift entitlements for masters and infra nodes. If you don't have different pool IDs, enter same pool ID as ' <code>rhsmPoolId</code> '		
<code>sshPublicKey</code>	Copy your SSH Public Key here		
<code>keyVaultSubscriptionId</code>	The Subscription ID of the subscription that contains the Key Vault		
<code>keyVaultResourceGroup</code>	The name of the Resource Group that contains the Key Vault		
<code>keyVaultName</code>	The name of the Key Vault you created		
<code>enableAzure</code>	Enable Azure Cloud Provider	true false	true
<code>aadClientId</code>	Azure Active Directory Client ID also known as Application ID for Service Principal		
<code>domainName</code>	Name of the custom domain name to use (if applicable). Set to "none" if not deploying fully private cluster		none
<code>masterClusterDnsType</code>	Domain type for OpenShift web console. 'default' will use DNS label of master infra public IP. 'custom' allows you to define your own name	default custom	default
<code>masterClusterDns</code>	The custom DNS name to use to access the OpenShift web console if you selected 'custom' for <code>masterClusterDnsType</code>		console.contoso.com
<code>routingSubDomainType</code>	If set to <code>nipio</code> , <code>routingSubDomain</code> will use <code>nip.io</code> . Use 'custom' if you have your own domain that you want to use for routing	<code>nipio</code> custom	<code>nipio</code>

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>routingSubDomain</code>	The wildcard DNS name you want to use for routing if you selected 'custom' for <code>routingSubDomainType</code>		apps.contoso.com
<code>virtualNetworkNewOrExisting</code>	Select whether to use an existing Virtual Network or create a new Virtual Network	existing new	new
<code>virtualNetworkResourceGroupName</code>	Name of the Resource Group for the new Virtual Network if you selected 'new' for <code>virtualNetworkNewOrExisting</code>		resourceGroup().name
<code>virtualNetworkName</code>	The name of the new Virtual Network to create if you selected 'new' for <code>virtualNetworkNewOrExisting</code>		openshiftvnet
<code>addressPrefixes</code>	Address prefix of the new virtual network		10.0.0.0/14
<code>masterSubnetName</code>	The name of the master subnet		mastersubnet
<code>masterSubnetPrefix</code>	CIDR used for the master subnet - needs to be a subset of the addressPrefix		10.1.0.0/16
<code>infraSubnetName</code>	The name of the infra subnet		infrasubnet
<code>infraSubnetPrefix</code>	CIDR used for the infra subnet - needs to be a subset of the addressPrefix		10.2.0.0/16
<code>nodeSubnetName</code>	The name of the node subnet		nodesubnet
<code>nodeSubnetPrefix</code>	CIDR used for the node subnet - needs to be a subset of the addressPrefix		10.3.0.0/16
<code>existingMasterSubnetReference</code>	Full reference to existing subnet for master nodes. Not needed if creating new vNet / Subnet		
<code>existingInfraSubnetReference</code>	Full reference to existing subnet for infra nodes. Not needed if creating new vNet / Subnet		

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>existingCnsSubnetReference</code>	Full reference to existing subnet for CNS nodes. Not needed if creating new vNet / Subnet		
<code>existingNodeSubnetReference</code>	Full reference to existing subnet for compute nodes. Not needed if creating new vNet / Subnet		
<code>masterClusterType</code>	Specify whether the cluster uses private or public master nodes. If private is chosen, the master nodes won't be exposed to the Internet via a public IP. Instead, it will use the private IP specified in the <code>masterPrivateClusterIp</code>	public private	public
<code>masterPrivateClusterIp</code>	If private master nodes are selected, then a private IP address must be specified for use by the internal load balancer for master nodes. This static IP must be within the CIDR block for the master subnet and not already in use. If public master nodes are selected, this value won't be used but must still be specified		10.1.0.200
<code>routerClusterType</code>	Specify whether the cluster uses private or public infra nodes. If private is chosen, the infra nodes won't be exposed to the Internet via a public IP. Instead, it will use the private IP specified in the <code>routerPrivateClusterIp</code>	public private	public
<code>routerPrivateClusterIp</code>	If private infra nodes are selected, then a private IP address must be specified for use by the internal load balancer for infra nodes. This static IP must be within the CIDR block for the infra subnet and not already in use. If public infra nodes are selected, this value won't be used but must still be specified		10.2.0.200

PROPERTY	DESCRIPTION	VALID OPTIONS	DEFAULT VALUE
<code>routingCertType</code>	Use custom certificate for routing domain or the default self-signed certificate - follow instructions in <b>Custom Certificates</b> section	selfsigned custom	selfsigned
<code>masterCertType</code>	Use custom certificate for master domain or the default self-signed certificate - follow instructions in <b>Custom Certificates</b> section	selfsigned custom	selfsigned

## Deploy using Azure CLI

### NOTE

The following command requires Azure CLI 2.0.8 or later. You can verify the CLI version with the `az --version` command. To update the CLI version, see [Install Azure CLI](#).

The following example deploys the OpenShift cluster and all related resources into a resource group named `openshiftrg`, with a deployment name of `myOpenShiftCluster`. The template is referenced directly from the GitHub repo, and a local parameters file named `azuredeploy.parameters.json` file is used.

```
az deployment group create -g openshiftrg --name myOpenShiftCluster \
    --template-uri https://raw.githubusercontent.com/Microsoft/openshift-container-
platform/master/azuredeploy.json \
    --parameters @./azuredeploy.parameters.json
```

The deployment takes at least 60 minutes to complete, based on the total number of nodes deployed and options configured. The Bastion DNS FQDN and URL of the OpenShift console prints to the terminal when the deployment finishes.

```
{
  "Bastion DNS FQDN": "bastiondns4hawllzaavu6g.eastus.cloudapp.azure.com",
  "OpenShift Console URL": "http://openshiftlb.eastus.cloudapp.azure.com/console"
}
```

If you don't want to tie up the command line waiting for the deployment to complete, add `--no-wait` as one of the options for the group deployment. The output from the deployment can be retrieved from the Azure portal in the deployment section for the resource group.

## Connect to the OpenShift cluster

When the deployment finishes, retrieve the connection from the output section of the deployment. Connect to the OpenShift console with your browser by using the **OpenShift Console URL**. You can also SSH to the Bastion host. Following is an example where the admin username is `clusteradmin` and the bastion public IP DNS FQDN is `bastiondns4hawllzaavu6g.eastus.cloudapp.azure.com`:

```
$ ssh clusteradmin@bastiondns4hawllzaavu6g.eastus.cloudapp.azure.com
```

## Clean up resources

Use the [az group delete](#) command to remove the resource group, OpenShift cluster, and all related resources when they're no longer needed.

```
az group delete --name openshifttrg
```

## Next steps

- [Post-deployment tasks](#)
- [Troubleshoot OpenShift deployment in Azure](#)
- [Getting started with OpenShift Container Platform](#)

# Configure prerequisites

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Before using the Marketplace offer to deploy a self-managed OpenShift Container Platform 3.11 cluster in Azure, a few prerequisites must be configured. Read the [OpenShift prerequisites](#) article for instructions to create an ssh key (without a passphrase), Azure key vault, key vault secret, and a service principal.

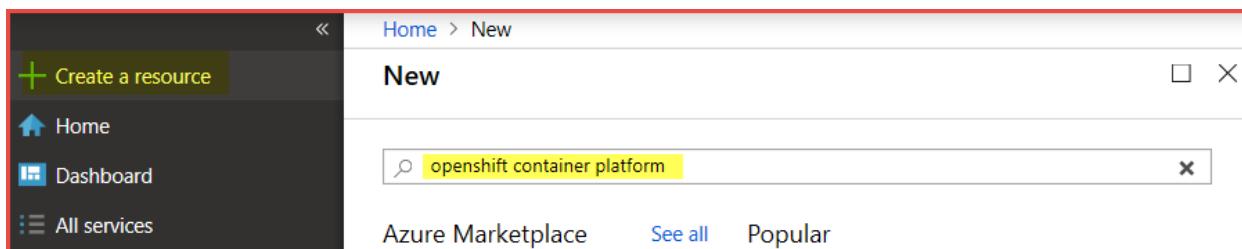
## Deploy using the Marketplace offer

The simplest way to deploy a self-managed OpenShift Container Platform 3.11 cluster into Azure is to use the Azure Marketplace offer.

This option is the simplest, but it also has limited customization capabilities. The Marketplace offer deploys OpenShift Container Platform 3.11.82 and includes the following configuration options:

- **Master Nodes:** Three (3) Master Nodes with configurable instance type.
- **Infra Nodes:** Three (3) Infra Nodes with configurable instance type.
- **Nodes:** The number of Nodes (between 1 and 9) and the instance type are configurable.
- **Disk Type:** Managed Disks are used.
- **Networking:** Support for new or existing Network and custom CIDR range.
- **CNS:** CNS can be enabled.
- **Metrics:** Hawkular Metrics can be enabled.
- **Logging:** EFK Logging can be enabled.
- **Azure Cloud Provider:** Enabled by default, can be disabled.

In the upper left of the Azure portal, click **Create a resource**, enter 'openshift container platform' into the search box and hit Enter.



The Results page will open with **Red Hat OpenShift Container Platform 3.11 Self-Managed** in the list.

Results		
NAME	PUBLISHER	CATEGORY
 Red Hat OpenShift Container Platform Self-Managed	Red Hat	Compute

Click the offer to view details of the offer. To deploy this offer, click **Create**. The UI to enter necessary parameters will appear. The first screen is the **Basics** blade.

**Red Hat OpenShift Container Platform Self-Managed**

Red Hat



Red Hat

[Create](#) [Save for later](#)

**This is the Self-Managed offer**

Read [Deployment Guide](#) for Azure Marketplace before deploying.

Red Hat OpenShift Container Platform helps organizations develop, deploy, and manage container-based applications seamlessly across physical, virtual, and public cloud infrastructures. OpenShift Container Platform brings application development and IT operations teams together in order to modernize applications, accelerate development processes, and deliver new services faster.

**OpenShift Highlights:**

- Simple to use with powerful tools for developers
- For traditional, stateful, and cloud-native applications
- Built on proven open source technologies including Red Hat Enterprise Linux, Kubernetes, and Docker
- Enterprise-grade security, compliance, and container management
- Expanded applications support with new and updated runtimes

This offering includes one bastion host, three master nodes, three infrastructure nodes, and a customizable number and size of application nodes.

- The number of application nodes is configurable between one and nine hosts with six options for the size of the virtual machines.
- The bastion host serves as a jump host for access to the OpenShift cluster nodes and system management.

**Useful Links**

- [OpenShift Container Platform Documentation](#)
- [OpenShift Container Platform Overview](#)
- [OpenShift Container Platform Reference Architecture for Azure](#)
- [Red Hat Cloud Access Program](#)
- [Red Hat Solutions On Azure](#)
- [Deploy OpenShift on Azure](#)

**OPENSHIFT CONTAINER PLATFORM**

Web Tier 1 > Add to Project

[Browse Catalog](#) [Deploy Image](#) [Import YAML / JSON](#)

Choose from web frameworks, databases, and other components to add content to your project.

Filter by keyword:  [Browse](#)

**Instant Apps**

 <a href="#">jenkins-ephemeral</a> INSTANTAPP JENKINS	 <a href="#">rs-java-openshift:1.0</a> BUILDER BUILDPACK JAVA MMAS
 <a href="#">jenkins-persistent</a> INSTANTAPP JENKINS	 <a href="#">rs-karaf-openshift:1.0</a> BUILDER BUILDPACK JBoss Karaf MMAS
 <a href="#">ruby-helloworld-sample</a> INSTANTAPP RUBY WHIZZ	 <a href="#">jboss-decisionserver62-openshift:1.2</a> BUILDER BUILDPACK JBoss Decisionserver MMAS
<a href="#">See all</a>	<a href="#">s2i-karaf2-camel-amq</a>

**xPaaS**

 <a href="#">s2i-karaf2-camel-amq</a> BUILDER BUILDPACK JBoss Decisionserver MMAS
---

## Basics

To get help on any of the input parameters, hover over the *i* next to the parameter name.

Enter values for the input parameters and click **OK**.

INPUT PARAMETER	PARAMETER DESCRIPTION
VM Admin User Name	The administrator user to be created on all VM instances
SSH Public Key for Admin User	SSH public key used to log into VM - must not have a passphrase
Subscription	Azure subscription to deploy cluster into
Resource Group	Create a new resource group or select an existing empty resource group for cluster resources
Location	Azure region to deploy cluster into

**Create Red Hat OpenShift Co... X**

**Basics** X

<b>1</b> Basics > Configure basic settings	<b>* VM Admin User Name</b> ⓘ <input type="text" value="clusteradmin"/>
<b>2</b> Infrastructure Settings > Configure Infrastructure Settings	<b>* SSH Public Key for VM Admin User</b> ⓘ <div style="border: 1px solid #ccc; padding: 5px; height: 40px; margin-top: 10px;"></div>
<b>3</b> OpenShift Container Platfor... > Configure OpenShift Container Pl...	<b>Subscription</b> <div style="border: 1px solid #ccc; padding: 5px; width: 100%; margin-bottom: 5px;"></div>
<b>4</b> Additional Settings > Additional Settings	<b>* Resource group</b> ⓘ <div style="border: 1px solid #ccc; padding: 5px; width: 100%; margin-bottom: 5px;">           (New) openshift-resourcegroup         </div> <a href="#">Create new</a>
<b>5</b> Summary > Red Hat OpenShift Container Plat...	<b>* Location</b> <div style="border: 1px solid #ccc; padding: 5px; width: 100%; margin-bottom: 5px;">           East US         </div>
<b>6</b> Buy >	

## Infrastructure Settings

Enter values for the input parameters and click OK.

INPUT PARAMETER	PARAMETER DESCRIPTION
OCP Cluster Name Prefix	Cluster Prefix used to configure hostnames for all nodes. Between 1 and 20 characters
Master Node Size	Accept the default VM size or click <b>Change size</b> to select a different VM size. Select appropriate VM size for your work load
Infrastructure Node Size	Accept the default VM size or click <b>Change size</b> to select a different VM size. Select appropriate VM size for your work load
Number of Application Nodes	Accept the default VM size or click <b>Change size</b> to select a different VM size. Select appropriate VM size for your work load
Application Node Size	Accept the default VM size or click <b>Change size</b> to select a different VM size. Select appropriate VM size for your work load
Bastion Host Size	Accept the default VM size or click <b>Change size</b> to select a different VM size. Select appropriate VM size for your work load

INPUT PARAMETER	PARAMETER DESCRIPTION
New or Existing Virtual Network	Create a new vNet (Default) or use an existing vNet
Choose Default CIDR Settings or customize IP Range (CIDR)	Accept default CIDR ranges or Select <b>Custom IP Range</b> and enter custom CIDR information. Default Settings will create vNet with CIDR of 10.0.0.0/14, master subnet with 10.1.0.0/16, infra subnet with 10.2.0.0/16, and compute and cns subnet with 10.3.0.0/16
Key Vault Resource Group Name	The name of the Resource Group that contains the Key Vault
Key Vault Name	The name of the Key Vault that contains the secret with the ssh private key. Only alphanumeric characters and dashes are allowed, and be between 3 and 24 characters
Secret Name	The name of the secret that contains the ssh private key. Only alphanumeric characters and dashes are allowed

**Create Red Hat OpenShift Co... X**

**Infrastructure Settings X**

<b>1 Basics</b> Done ✓	<b>* OCP Cluster Name Prefix</b> <input type="text" value="ocpcluster"/>
<b>2 Infrastructure Settings</b> > Configure Infrastructure Settings	<b>* Master Node Size</b> <b>3x Standard D4s v3</b> 4 vcpus, 16 GB memory <a href="#">Change size</a>
<b>3 OpenShift Container Platfor...</b> > Configure OpenShift Container Pl...	<b>* Infrastructure Node Size</b> <b>3x Standard E2s v3</b> 2 vcpus, 16 GB memory <a href="#">Change size</a>
<b>4 Additional Settings</b> > Additional Settings	<b>Number of Application Nodes</b> <input type="text" value="3"/> ✓
<b>5 Summary</b> > Red Hat OpenShift Container Plat...	<b>* Application Node Size</b> <b>3x Standard D2s v3</b> 2 vcpus, 8 GB memory <a href="#">Change size</a>
<b>6 Buy</b> >	<b>* Bastion Host Size</b> <b>1x Standard DS2 v2</b> 2 vcpus, 7 GB memory <a href="#">Change size</a>
	<b>New or Existing Virtual Network</b> <input checked="" type="radio" value="Default (New)"/> <input type="radio" value="Existing"/> Default (New) Existing
	Choosing the default or to customize the Virtual Network <input checked="" type="radio" value="Default Settings"/> <input type="radio" value="Custom IP Range"/> Default Settings Custom IP Range
	<b>* Key Vault Resource Group Name</b> <input type="text" value="keyvaultrg"/> ✓
	<b>* Key Vault Name</b> <input type="text" value="keyvault"/> ✓
	<b>* Secret Name</b> <input type="text" value="sshprivatekey"/>

## Change size

To select a different VM size, click **Change size**. The VM selection window will open. Select the VM size you want and click **Select**.

Showing 4 VM sizes. | Subscription: [REDACTED] | Region: East US | Current size: Standard\_D4s\_v3

VM SIZE	OFFERING	FAMILY	VCPUS	RAM (GB)	DATA DISKS	MAX IOPS	TEMPORARY STORA...	PREMIUM DISK SUP...	COST/MONTH (ESTI...)
D2s_v3	Standard	General purpose	2	8	4	3200	16 GB	Yes	[REDACTED]
D4s_v3	Standard	General purpose	4	16	8	6400	32 GB	Yes	[REDACTED]
E2s_v3	Standard	Memory optimized	2	16	4	3200	32 GB	Yes	[REDACTED]
E4s_v3	Standard	Memory optimized	4	32	8	6400	64 GB	Yes	[REDACTED]

## Existing Virtual Network

INPUT PARAMETER	PARAMETER DESCRIPTION
Existing Virtual Network Name	Name of the existing vNet
Subnet name for master nodes	Name of existing subnet for master nodes. Needs to contain at least 16 IP addresses and follow RFC 1918
Subnet name for infra nodes	Name of existing subnet for infra nodes. Needs to contain at least 32 IP addresses and follow RFC 1918
Subnet name for compute and cns nodes	Name of existing subnet for compute and cns nodes. Needs to contain at least 32 IP addresses and follow RFC 1918
Resource Group for the existing Virtual Network	Name of resource group that contains the existing vNet

New or Existing Virtual Network ⓘ

Default (New)  Existing

\* Existing Virtual Network Name  
ocpvnet ✓

\* Subnet name for master nodes ⓘ  
mastersubnet ✓

\* Subnet name for infra nodes ⓘ  
infrasubnet ✓

\* Subnet name for compute and cns nodes ⓘ  
nodesubnet ✓

\* Resource Group for the existing Virtual Network ⓘ  
vnetresourcegroup ✓

## Custom IP Range

INPUT PARAMETER	PARAMETER DESCRIPTION
Address Range for the Virtual Network	Custom CIDR for the vNet
Address Range for the subnet containing the master nodes	Custom CIDR for master subnet
Address Range for the subnet containing the infrastructure nodes	Custom CIDR for infrastructure subnet
Address Range for subnet containing the compute and cns nodes	Custom CIDR for the compute and cns nodes

Choosing the default or to customize the Virtual Network [?](#)

[Default Settings](#) [Custom IP Range](#)

\* Address Range for the VirtualNetwork (default is 10.0.0.0/14) [?](#)  
10.0.0.0/16 ✓

\* Address Range for the subnet containing the master, infra, and cns nodes (default is 10.1.0.0/16) [?](#)  
10.0.10.0/24 ✓

\* Address Range for the subnet containing all the infrastructure nodes (default is 10.2.0.0/16) [?](#)  
10.0.20.0/24 ✓

\* Address Range for the subnet containing all the cns and compute nodes (default is 10.3.0.0/16) [?](#)  
10.0.30.0/24 ✓

## OpenShift Container Platform 3.11

Enter values for the Input Parameters and click OK

INPUT PARAMETER	PARAMETER DESCRIPTION
OpenShift Admin User Password	Password for the initial OpenShift user. This user will also be the cluster admin
Confirm OpenShift Admin User Password	Retype the OpenShift Admin User Password
Red Hat Subscription Manager User Name	User Name to access your Red Hat Subscription or Organization ID. This credential is used to register the RHEL instance to your subscription and will not be stored by Microsoft or Red Hat
Red Hat Subscription Manager User Password	Password to access your Red Hat Subscription or Activation Key. This credential is used to register the RHEL instance to your subscription and will not be stored by Microsoft or Red Hat
Red Hat Subscription Manager OpenShift Pool ID	Pool ID that contains OpenShift Container Platform entitlement. Ensure you have enough entitlements of OpenShift Container Platform for the installation of the cluster
Red Hat Subscription Manager OpenShift Pool ID for Broker / Master Nodes	Pool ID that contains OpenShift Container Platform entitlements for Broker / Master Nodes. Ensure you have enough entitlements of OpenShift Container Platform for the installation of the cluster. If not using broker / master pool ID, enter the pool ID for Application Nodes
Configure Azure Cloud Provider	Configure OpenShift to use Azure Cloud Provider. Necessary if using Azure disk attach for persistent volumes. Default is Yes

INPUT PARAMETER	PARAMETER DESCRIPTION
Azure AD Service Principal Client ID GUID	Azure AD Service Principal Client ID GUID - also known as AppID. Only needed if Configure Azure Cloud Provider set to Yes
Azure AD Service Principal Client ID Secret	Azure AD Service Principal Client ID Secret. Only needed if Configure Azure Cloud Provider set to Yes

**Create Red Hat OpenShift Co... X**

- 1 Basics** Done ✓
- 2 Infrastructure Settings** Done ✓
- 3 OpenShift Container Platfor...** > Configure OpenShift Container Pl...
- 4 Additional Settings** > Additional Settings
- 5 Summary** > Red Hat OpenShift Container Plat...
- 6 Buy** >

**OpenShift Container Plat...** □ X

\* OpenShift Admin User Password ⓘ  ✓

\* Confirm OpenShift Admin User Password  ✓

\* Red Hat Subscription Manager User Name ⓘ  ✓

\* Red Hat Subscription Manager User Password ⓘ  ✓

\* Red Hat Subscription Manager OpenShift Pool ID ⓘ  ✓

\* Red Hat Subscription Manager OpenShift Pool ID for Broker / Master Nodes ⓘ  ✓

Configure Azure Cloud Provider ⓘ   
  Yes  No

\* Azure AD Service Principal Client ID GUID ⓘ  ✓

\* Azure AD Service Principal Client ID Secret ⓘ  ✓

## Additional Settings

The Additional Settings blade allows the configuration of CNS for glusterfs storage, Logging, Metrics, and Router Sub domain. The default won't install any of these options and will use nip.io as the router sub domain for testing purposes. Enabling CNS will install three additional compute nodes with three additional attached disks that will host glusterfs pods.

Enter values for the Input Parameters and click OK

INPUT PARAMETER	PARAMETER DESCRIPTION
-----------------	-----------------------

INPUT PARAMETER	PARAMETER DESCRIPTION
Configure Container Native Storage (CNS)	Installs CNS in the OpenShift cluster and enable it as storage. Will be default if Azure Provider is disabled
Configure Cluster Logging	Installs EFK logging functionality into the cluster. Size infra nodes appropriately to host EFK pods
Configure Metrics for the Cluster	Installs Hawkular metrics into the OpenShift cluster. Size infra nodes appropriately to host Hawkular metrics pods
Default Router Sub domain	Select nipio for testing or custom to enter your own sub domain for production

Create Red Hat OpenShift Co... X Additional Settings □ X

The screenshot shows the 'Create Red Hat OpenShift Container Platform' wizard. Step 4, 'Additional Settings', is currently selected. On the left, a vertical navigation bar lists steps 1 through 6: 1. Basics (Done), 2. Infrastructure Settings (Done), 3. OpenShift Container Platform (Done), 4. Additional Settings (selected), 5. Summary, and 6. Buy. Step 4 contains three configuration options:

- Configure Container Native Storage (CNS) - A toggle switch set to 'Yes'.
- Configure Cluster Logging - A toggle switch set to 'No'.
- Configure Metrics for the Cluster - A toggle switch set to 'Yes'.

Below these is a dropdown menu for 'Default Router Subdomain' set to 'nipio'. The 'Buy' button at the bottom right of the step 4 panel is highlighted in blue.

#### Additional Settings - Extra Parameters

INPUT PARAMETER	PARAMETER DESCRIPTION
(CNS) Node Size	Accept the default node size or select <b>Change size</b> to select a new VM size
Enter your custom subdomain	The custom routing domain to be used for exposing applications via the router on the OpenShift cluster. Be sure to create the appropriate wildcard DNS entry]

Create Red Hat OpenShift Co... X Additional Settings □ X

1 Basics Done	✓
2 Infrastructure Settings Done	✓
3 OpenShift Container Platfor... Done	✓
4 Additional Settings Additional Settings >	
5 Summary Red Hat OpenShift Container Plat...	>
6 Buy	>

Configure Container Native Storage (CNS) ⓘ  
 Yes  No

\* CNS Node Size ⓘ  
**3x Standard E4s v3**  
4 vcpus, 32 GB memory  
[Change size](#)

Configure Cluster Logging ⓘ  
 Yes  No

Configure Metrics for the Cluster ⓘ  
 Yes  No

Default Router Subdomain ⓘ  
custom

\* Enter your custom subdomain ⓘ  
apps.contoso.com ✓

## Summary

Validation occurs at this stage to check core quota is sufficient to deploy the total number of VMs selected for the cluster. Review all the parameters that were entered. If the inputs are acceptable, click **OK** to continue.

**Create Red Hat OpenShift Container Platform**

Summary	
<b>1</b> Basics      ✓ Done	
<b>2</b> Infrastructure Settings      ✓ Done	
<b>3</b> OpenShift Container Platform Settings      ✓ Done	
<b>4</b> Additional Settings      ✓ Done	
<b>5</b> Summary > Red Hat OpenShift Container Platform Settings	
<b>6</b> Buy >	

**Validation passed**

**Basics**

Subscription	[REDACTED]
Resource group	openshift-resourcegroup
Location	East US

**VM Admin User Name** clusteradmin  
**SSH Public Key for VM Admin** [REDACTED]

**Infrastructure Settings**

OCP Cluster Name Prefix	ocpcluster
Master Node Size	Standard D4s v3
Infrastructure Node Size	Standard D4s v3
Number of Application Nodes	3
Application Node Size	Standard D4s v3
Bastion Host Size	Standard DS2 v2
New or Existing Virtual Network	Default (New)
Choosing the default or to customize	Default Settings
Key Vault Resource Group Name	[REDACTED]
Key Vault Name	[REDACTED]
Secret Name	sshprivatekey

**OpenShift Container Platform Settings**

OpenShift Admin User Password	*****
Red Hat Subscription Manager Username	rhmusername
Red Hat Subscription Manager Password	*****
Red Hat Subscription Manager Client ID	8abcd12345e6f7890123abdbe01a000a
Red Hat Subscription Manager Client Secret	8abcd12345e6f7890123abdbe01a000b
Configure Azure Cloud Provider	Yes
Azure AD Service Principal Client ID	abcd1234-5678-9ef0-89cb-d6da4f6cde12
Azure AD Service Principal Client Secret	*****

**Additional Settings**

Configure Container Native Subdomain	Yes
CNS Node Size	Standard E4s v3
Configure Cluster Logging	No
Configure Metrics for the Cluster	No
Default Router Subdomain	nipio

## Buy

Confirm contact information on the Buy page and click **Purchase** to accept the terms of use and start deployment of the OpenShift Container Platform cluster.

**Create Red Hat OpenShift Co... X**

Create		
<b>1</b>	Basics	✓
Done		
<b>2</b>	Infrastructure Settings	✓
Done		
<b>3</b>	OpenShift Container Plat...	✓
Done		
<b>4</b>	Additional Settings	✓
Done		
<b>5</b>	Summary	✓
Red Hat OpenShift Container Plat...		
<b>6</b>	Buy	>

Red Hat OpenShift Container Platform Self-Managed by Red Hat  
[Terms of use](#) | [privacy policy](#)

Deploying this template will result in various actions being performed, which may include the deployment of one or more Azure resources or Marketplace offerings and/or transmission of the information you provided as part of the deployment process to one or more parties, as specified in the template. You are responsible for reviewing the text of the template to determine which actions will be performed and which resources or offerings will be deployed, and for locating and reviewing the pricing and legal terms associated with those resources or offerings.

Current retail prices for Azure resources are set forth [here](#) and may not reflect discounts applicable to your Azure subscription.

Prices for Marketplace offerings are set forth [here](#), and the legal terms associated with any Marketplace offering may be found in the Azure portal; both are subject to change at any time prior to deployment.

Neither subscription credits nor monetary commitment funds may be used to purchase non-Microsoft offerings. These purchases are billed separately. If any Microsoft products are included in a Marketplace offering (e.g., Windows Server or SQL Server), such products are licensed by Microsoft and not by any third party.

**Template deployment is intended for advanced users only.** If you are uncertain which actions will be performed by this template, which resources or offerings will be deployed, or what prices or legal terms pertain to those resources or offerings, do not deploy this template.

**Terms of use**

By clicking "Create", I (a) agree to the legal terms and privacy statement(s) provided above as well as the legal terms and privacy statement(s) associated with each Marketplace offering that will be deployed using this template, if any; (b) authorize Microsoft to charge or bill my current payment method for the fees associated with my use of the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); (c) agree that Microsoft may share my contact information and transaction details with any third-party sellers of the offering(s); and (d) give Microsoft permission to share my contact information so that the provider of the template can contact me regarding this product and related products. Microsoft assumes no responsibility for any actions performed by third-party templates and does not provide rights for third-party products or services. See the [Azure Marketplace Terms](#) for additional terms.

**By clicking Create, you give Microsoft permission to use or share your account information so that the provider or Microsoft can contact you regarding this product and related products.**

Name:

\* Preferred e-mail address:

\* Preferred phone number:

**Create**

## Connect to the OpenShift cluster

When the deployment finishes, retrieve the connection from the output section of the deployment. Connect to the OpenShift console with your browser by using the **OpenShift Console URL**. You can also SSH to the Bastion host. Following is an example where the admin username is clusteradmin and the bastion public IP DNS FQDN is bastiondns4hawllzaavu6g.eastus.cloudapp.azure.com:

```
$ ssh clusteradmin@bastiondns4hawllzaavu6g.eastus.cloudapp.azure.com
```

## Clean up resources

Use the [az group delete](#) command to remove the resource group, OpenShift cluster, and all related resources when they're no longer needed.

```
az group delete --name openshifttrg
```

## Next steps

- [Post-deployment tasks](#)
- [Troubleshoot OpenShift deployment in Azure](#)
- [Getting started with OpenShift Container Platform](#)
-

# Deploy OpenShift Container Platform or OKD to Azure Stack Hub

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

[OpenShift](#) can be deployed in Azure Stack Hub. There are some key differences between Azure and Azure Stack Hub so deployment will differ slightly and capabilities will also differ slightly.

Currently, the Azure Cloud Provider doesn't work in Azure Stack Hub. You won't be able to use disk attach for persistent storage in Azure Stack Hub. Instead, you can configure other storage options such as NFS, iSCSI, and GlusterFS. Or, you can enable CNS and use GlusterFS for persistent storage. If CNS is enabled, three more nodes will be deployed with storage for GlusterFS usage.

## Deploy OpenShift 3.x On Azure Stack Hub

You can use one of several methods to deploy OpenShift Container Platform or OKD in Azure Stack Hub:

- You can manually deploy the necessary Azure infrastructure components and then follow the [OpenShift Container Platform documentation](#) or [OKD documentation](#).
- You can also use an existing [Azure Resource Manager template](#) that simplifies the deployment of the OpenShift Container Platform cluster.
- You can also use an existing [Azure Resource Manager template](#) that simplifies the deployment of the OKD cluster.

If using the Azure Resource Manager template, select the proper branch (azurestack-release-3.x). The templates for Azure won't work as the API versions are different between Azure and Azure Stack Hub. The RHEL image reference is currently hard-coded as a variable in the azuredeploy.json file and will need to be changed to match your image.

```
"imageReference": {  
    "publisher": "Redhat",  
    "offer": "RHEL-OCP",  
    "sku": "7-4",  
    "version": "latest"  
}
```

For all options, a Red Hat subscription is required. During the deployment, the Red Hat Enterprise Linux instance is registered to the Red Hat subscription and attached to the Pool ID that contains the entitlements for OpenShift Container Platform. Make sure you have a valid Red Hat Subscription Manager (RHSM) username, password, and Pool ID. Alternatively, you can use an Activation Key, Org ID, and Pool ID. You can verify this information at <https://access.redhat.com>.

### Azure Stack Hub prerequisites

An RHEL image (OpenShift Container Platform) or CentOS image (OKD) needs to be added to your Azure Stack Hub environment to deploy an OpenShift cluster. Contact your Azure Stack Hub cloud operator to add these images. Instructions can be found here:

- [Add and remove a custom VM image to Azure Stack Hub](#)
- [Azure Marketplace items available for Azure Stack Hub](#)

- Offer a Red Hat-based virtual machine for Azure Stack Hub

## Deploy by using the OpenShift Container Platform or OKD Azure Resource Manager template

To deploy by using the Azure Resource Manager template, you use a parameters file to supply the input parameters. To further customize the deployment, fork the GitHub repo and change the appropriate items.

Some common customization options include, but aren't limited to:

- Bastion VM size (variable in `azuredploy.json`)
- Naming conventions (variables in `azuredploy.json`)
- OpenShift cluster specifics, modified via hosts file (`deployOpenShift.sh`)
- RHEL image reference (variable in `azuredploy.json`)

For the steps to deploy using the Azure CLI, follow the appropriate section in the [OpenShift Container Platform](#) section or the [OKD](#) section.

## Deploy OpenShift 4.x On Azure Stack Hub

Red Hat manages the Red Hat Enterprise Linux CoreOS (RHCOS) image for OpenShift 4.x. The deployment process gets the image from a Red Hat endpoint. As a result, the user (tenant) doesn't need to get an image from the Azure Stack hub Marketplace.

You can follow the steps in the OpenShift documentation at [Installing a cluster on Azure Stack Hub using ARM templates](#).

### WARNING

If you have an issue with OpenShift, please contact Red Hat for support.

## Next steps

- [Post-deployment tasks](#)
- [Troubleshoot OpenShift deployment in Azure](#)

# Post-deployment tasks

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

After you deploy an OpenShift cluster, you can configure additional items. This article covers:

- How to configure single sign-on by using Azure Active Directory (Azure AD)
- How to configure Azure Monitor logs to monitor OpenShift
- How to configure metrics and logging
- How to install Open Service Broker for Azure (OSBA)

## Configure single sign-on by using Azure Active Directory

To use Azure Active Directory for authentication, first you need to create an Azure AD app registration. This process involves two steps: creating the app registration, and configuring permissions.

### Create an app registration

These steps use the Azure CLI to create the app registration, and the GUI (portal) to set the permissions. To create the app registration, you need the following five pieces of information:

- Display name: App registration name (for example, OCPAzureAD)
- Home page: OpenShift console URL (for example,  
`https://masterdns343khde.westus.cloudapp.azure.com/console`)
- Identifier URI: OpenShift console URL (for example,  
`https://masterdns343khde.westus.cloudapp.azure.com/console`)
- Reply URL: Master public URL and the app registration name (for example,  
`https://masterdns343khde.westus.cloudapp.azure.com/oauth2callback/OCPAzureAD`)
- Password: Secure password (use a strong password)

The following example creates an app registration by using the preceding information:

```
az ad app create --display-name OCPAzureAD --homepage  
https://masterdns343khde.westus.cloudapp.azure.com/console --reply-urls  
https://masterdns343khde.westus.cloudapp.azure.com/oauth2callback/hwocpadint --identifier-uris  
https://masterdns343khde.westus.cloudapp.azure.com/console --password {Strong Password}
```

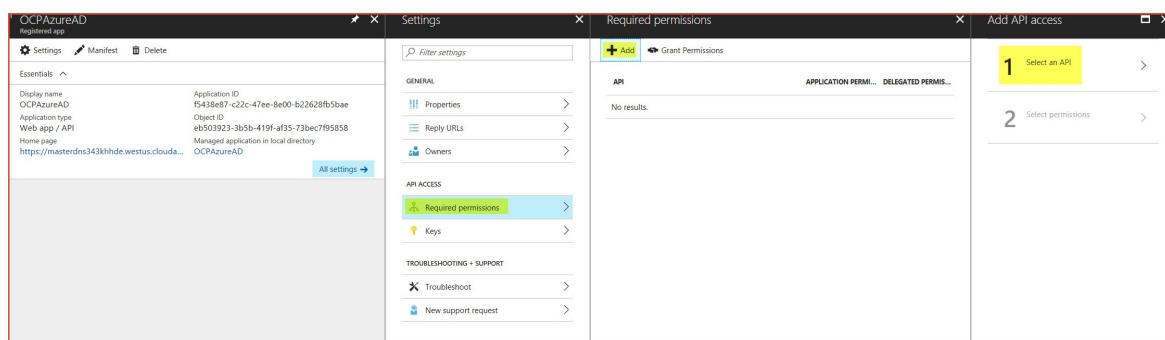
If the command is successful, you get a JSON output similar to:

```
{
  "appId": "12345678-ca3c-427b-9a04-ab12345cd678",
  "appPermissions": null,
  "availableToOtherTenants": false,
  "displayName": "OCPAzureAD",
  "homepage": "https://masterdns343khhde.westus.cloudapp.azure.com/console",
  "identifierUris": [
    "https://masterdns343khhde.westus.cloudapp.azure.com/console"
  ],
  "objectId": "62cd74c9-42bb-4b9f-b2b5-b6ee88991c80",
  "objectType": "Application",
  "replyUrls": [
    "https://masterdns343khhde.westus.cloudapp.azure.com/oauth2callback/OCPAzureAD"
  ]
}
```

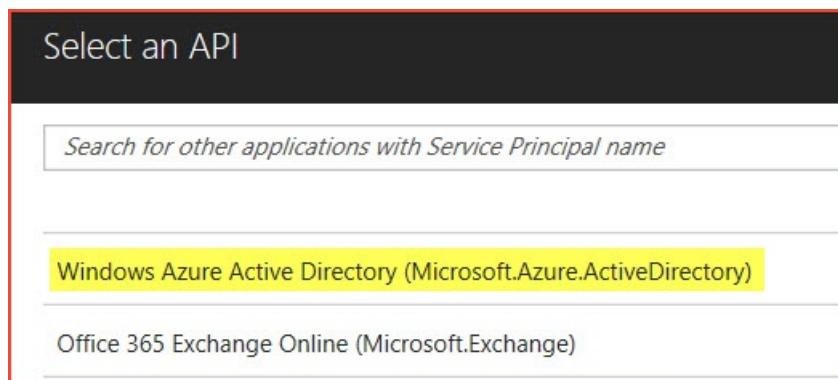
Take note of the appId property returned from the command for a later step.

In the Azure portal:

1. Select **Azure Active Directory > App Registration**.
2. Search for your app registration (for example, OCPAzureAD).
3. In the results, click the app registration.
4. Under **Settings**, select **Required permissions**.
5. Under **Required Permissions**, select **Add**.



6. Click Step 1: Select API, and then click **Windows Azure Active Directory (Microsoft.Azure.ActiveDirectory)**. Click **Select** at the bottom.



7. On Step 2: Select Permissions, select **Sign in and read user profile** under **Delegated Permissions**, and then click **Select**.

## Enable Access

□ X

APPLICATION PERMISSIONS	REQUIRES ADMIN
Read directory data	✓ Yes
Read and write domains	✓ Yes
Read and write directory data	✓ Yes
Read and write devices	✓ Yes
Read all hidden memberships	✓ Yes
Manage apps that this app creates or owns	✓ Yes
Read and write all applications	✓ Yes
Read and write domains	✓ Yes
DELEGATED PERMISSIONS	REQUIRES ADMIN
Access the directory as the signed-in user	✗ No
Read directory data	✓ Yes
Read and write directory data	✓ Yes
Read and write all groups	✓ Yes
Read all groups	✓ Yes
Read all users' full profiles	✓ Yes
Read all users' basic profiles	✗ No
Sign in and read user profile	✗ No
Read hidden memberships	✓ Yes

8. Select Done.

### Configure OpenShift for Azure AD authentication

To configure OpenShift to use Azure AD as an authentication provider, the /etc/origin/master/master-config.yaml file must be edited on all master nodes.

Find the tenant ID by using the following CLI command:

```
az account show
```

In the yaml file, find the following lines:

```
oauthConfig:
  assetPublicURL: https://masterdns343khhde.westus.cloudapp.azure.com/console/
  grantConfig:
    method: auto
  identityProviders:
    - challenge: true
      login: true
      mappingMethod: claim
      name: htpasswd_auth
      provider:
        apiVersion: v1
        file: /etc/origin/master/htpasswd
        kind: HTPasswdPasswordIdentityProvider
```

Insert the following lines immediately after the preceding lines:

```
- name: <App Registration Name>
  challenge: false
  login: true
  mappingMethod: claim
  provider:
    apiVersion: v1
    kind: OpenIDIdentityProvider
    clientID: <appId>
    clientSecret: <Strong Password>
    claims:
      id:
      - sub
    preferredUsername:
      - unique_name
    name:
      - name
    email:
      - email
  urls:
    authorize: https://login.microsoftonline.com/<tenant Id>/oauth2/authorize
    token: https://login.microsoftonline.com/<tenant Id>/oauth2/token
```

Make sure the text aligns correctly under identityProviders. Find the tenant ID by using the following CLI command: `az account show`

Restart the OpenShift master services on all master nodes:

```
sudo /usr/local/bin/master-restart api
sudo /usr/local/bin/master-restart controllers
```

In the OpenShift console, you now see two options for authentication: htpasswd\_auth and [App Registration].

## Monitor OpenShift with Azure Monitor logs

There are three ways to add the Log Analytics agent to OpenShift.

- Install the Log Analytics agent for Linux directly on each OpenShift node
- Enable Azure Monitor VM Extension on each OpenShift node
- Install the Log Analytics agent as an OpenShift daemon-set

Read the full [instructions](#) for more details.

## Configure metrics and logging

Based on the branch, the Azure Resource Manager templates for OpenShift Container Platform and OKD may provide input parameters for enabling metrics and logging as part of the installation.

The OpenShift Container Platform Marketplace offer also provides an option to enable metrics and logging during cluster installation.

If metrics / logging wasn't enabled during the installation of the cluster, they can easily be enabled after the fact.

### Azure Cloud Provider in use

SSH to the bastion node or first master node (based on template and branch in use) using the credentials provided during deployment. Issue the following command:

```
ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/openshift-metrics/config.yml \
-e openshift_metrics_install_metrics=True \
-e openshift_metrics_cassandra_storage_type=dynamic

ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/openshift-logging/config.yml \
-e openshift_logging_install_logging=True \
-e openshift_logging_es_pvc_dynamic=true
```

### Azure Cloud Provider not in use

```
ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/openshift-metrics/config.yml \
-e openshift_metrics_install_metrics=True

ansible-playbook /usr/share/ansible/openshift-ansible/playbooks/openshift-logging/config.yml \
-e openshift_logging_install_logging=True
```

## Install Open Service Broker for Azure (OSBA)

Open Service Broker for Azure, or OSBA, lets you provision Azure Cloud Services directly from OpenShift. OSBA in an Open Service Broker API implementation for Azure. The Open Service Broker API is a spec that defines a common language for cloud providers that cloud native applications can use to manage cloud services without lock-in.

To install OSBA on OpenShift, follow the instructions located here: <https://github.com/Azure/open-service-broker-azure#openshift-project-template>.

#### NOTE

Only complete the steps in the OpenShift Project Template section and not the entire Installing section.

## Next steps

- [Getting started with OpenShift Container Platform](#)

# Troubleshoot OpenShift Container Platform 3.11 deployment in Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

If the OpenShift cluster doesn't deploy successfully, the Azure portal will provide error output. The output may be difficult to read which makes it difficult to identify the problem. Quickly scan this output for exit code 3, 4 or 5. The following provides information on these three exit codes:

- Exit code 3: Your Red Hat Subscription User Name / Password or Organization ID / Activation Key is incorrect
- Exit code 4: Your Red Hat Pool ID is incorrect or there are no entitlements available
- Exit code 5: Unable to provision Docker Thin Pool Volume

For all other exit codes, connect to the host(s) via ssh to view the log files.

## OpenShift Container Platform 3.11

SSH to the ansible playbook host. For the template or the Marketplace offer, use the bastion host. From the bastion, you can SSH to all other nodes in the cluster (master, infra, CNS, compute). You'll need to be root to view the log files. Root is disabled for SSH access by default so don't use root to SSH to other nodes.

## OKD

SSH to the ansible playbook host. For the OKD template (version 3.9 and earlier), use the master-0 host. For the OKD template (version 3.10 and later), use the bastion host. From the ansible playbook host, you can SSH to all other nodes in the cluster (master, infra, CNS, compute). You'll need to be root (sudo su -) to view the log files. Root is disabled for SSH access by default so don't use root to SSH to other nodes.

## Log files

The log files (stderr and stdout) for the host preparation scripts are located in

`/var/lib/waagent/custom-script/download/0` on all hosts. If an error occurred during the preparation of the host, view these log files to determine the error.

If the preparation scripts ran successfully, then the log files in the `/var/lib/waagent/custom-script/download/1` directory of the ansible playbook host will need to be examined. If the error occurred during the actual installation of OpenShift, the stdout file will display the error. Use this information to contact Support for further assistance.

Example output

```

TASK [openshift_storage_glusterfs : Load heketi topology] ****
fatal: [mycluster-master-0]: FAILED! => {"changed": true, "cmd": ["oc", "--config=/tmp/openshift-glusterfs-ansible-IbhnUM/admin.kubeconfig", "rsh", "--namespace=glusterfs", "deploy-heketi-storage-1-d9x15", "heketi-cli", "-s", "http://localhost:8080", "--user", "admin", "--secret", "VuoJURT0/96E42Vv8+XHfsFpSS8R20rH10iMs3OqARQ=", "topology", "load", "--json=/tmp/openshift-glusterfs-ansible-IbhnUM/topology.json", ">&1"], "delta": "0:00:21.477831", "end": "2018-05-20 02:49:11.912899", "failed": true, "failed_when_result": true, "rc": 0, "start": "2018-05-20 02:48:50.435068", "stderr": "", "stderr_lines": [], "stdout": "Creating cluster ... ID: 794b285745b1c5d7089e1c5729ec7cd2\n\tAllowing file volumes on cluster.\n\tAllowing block volumes on cluster.\n\tCreating node mycluster-cns-0 ... ID: 45f1a3bfc20a4196e59ebb567e0e02b4\n\t\tAdding device /dev/sdd ... OK\n\t\tAdding device /dev/sde ...\nOK\n\t\tAdding device /dev/sdf ... OK\n\t\tCreating node mycluster-cns-1 ... ID: 596f80d7bbd78a1ea548930f23135131\n\t\tAdding device /dev/sdc ... Unable to add device: Unable to execute command on glusterfs-storage-4zc42: Device /dev/sdc excluded by a filter.\n\t\tAdding device /dev/sde ... OK\n\t\tCreating node mycluster-cns-2 ... ID: 42c0170aa2799559747622acceba2e3f\n\t\tAdding device /dev/sde ... OK\n\t\tAdding device /dev/sdf ...\nOK\n\t\tAdding device /dev/sdd ... OK", "stdout_lines": ["Creating cluster ... ID: 794b285745b1c5d7089e1c5729ec7cd2", "\tAllowing file volumes on cluster.", "\tAllowing block volumes on cluster.", "\tCreating node mycluster-cns-0 ... ID: 45f1a3bfc20a4196e59ebb567e0e02b4", "\t\tAdding device /dev/sdd ... OK", "\t\tAdding device /dev/sde ... OK", "\t\tAdding device /dev/sdf ... OK", "\tCreating node mycluster-cns-1 ... ID: 596f80d7bbd78a1ea548930f23135131", "\t\tAdding device /dev/sdc ... Unable to add device: Unable to execute command on glusterfs-storage-4zc42: Device /dev/sdc excluded by a filter.", "\t\tAdding device /dev/sde ... OK", "\t\tAdding device /dev/sdd ... OK", "\t\tCreating node mycluster-cns-2 ... ID: 42c0170aa2799559747622acceba2e3f", "\t\tAdding device /dev/sde ... OK", "\t\tAdding device /dev/sdf ... OK", "\t\tAdding device /dev/sdd ... OK"]}

PLAY RECAP ****
mycluster-cns-0      : ok=146   changed=57    unreachable=0    failed=0
mycluster-cns-1      : ok=146   changed=57    unreachable=0    failed=0
mycluster-cns-2      : ok=146   changed=57    unreachable=0    failed=0
mycluster-infra-0    : ok=143   changed=55    unreachable=0    failed=0
mycluster-infra-1    : ok=143   changed=55    unreachable=0    failed=0
mycluster-infra-2    : ok=143   changed=55    unreachable=0    failed=0
mycluster-master-0   : ok=502   changed=198   unreachable=0    failed=1
mycluster-master-1   : ok=348   changed=140   unreachable=0    failed=0
mycluster-master-2   : ok=348   changed=140   unreachable=0    failed=0
mycluster-node-0     : ok=143   changed=55    unreachable=0    failed=0
mycluster-node-1     : ok=143   changed=55    unreachable=0    failed=0
localhost            : ok=13    changed=0     unreachable=0    failed=0

INSTALLER STATUS ****
Initialization          : Complete (0:00:39)
Health Check           : Complete (0:00:24)
etcd Install           : Complete (0:01:24)
Master Install          : Complete (0:14:59)
Master Additional Install: Complete (0:01:10)
Node Install            : Complete (0:10:58)
GlusterFS Install       : In Progress (0:03:33)
This phase can be restarted by running: playbooks/openshift-glusterfs/config.yml

Failure summary:

1. Hosts: mycluster-master-0
Play: Configure GlusterFS
Task: Load heketi topology
Message: Failed without returning a message.

```

The most common errors during installation are:

1. Private key has passphrase
2. Key vault secret with private key wasn't created correctly
3. Service principal credentials were entered incorrectly
4. Service principal doesn't have contributor access to the resource group

### Private Key has a passphrase

You'll see an error that permission was denied for ssh. ssh to the ansible playbook host to check for a passphrase on the private key.

### **Key vault secret with private key wasn't created correctly**

The private key is copied into the ansible playbook host - `~/.ssh/id_rsa`. Confirm this file is correct. Test by opening an SSH session to one of the cluster nodes from the ansible playbook host.

### **Service principal credentials were entered incorrectly**

When providing the input to the template or Marketplace offer, the incorrect information was provided. Make sure you use the correct appId (clientId) and password (clientSecret) for the service principal. Verify by issuing the following azure cli command.

```
az login --service-principal -u <client id> -p <client secret> -t <tenant id>
```

### **Service principal doesn't have contributor access to the resource group**

If the Azure cloud provider is enabled, then the service principal used must have contributor access to the resource group. Verify by issuing the following azure cli command.

```
az group update -g <openshift resource group> --set tags.sptest=test
```

## Additional tools

For some errors, you can also use the following commands to get more information:

1. `systemctl status <service>`
2. `journalctl -xe`

# Deploy OKD in Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You can use one of two ways to deploy OKD (formerly OpenShift Origin) in Azure:

- You can manually deploy all the necessary Azure infrastructure components, and then follow the [OKD documentation](#).
- You can also use an existing [Resource Manager template](#) that simplifies the deployment of the OKD cluster.

## Deploy using the OKD template

To deploy using the Resource Manager template, you use a parameters file to supply the input parameters. To further customize the deployment, fork the GitHub repo and change the appropriate items.

Some common customization options include, but aren't limited to:

- Bastion VM size (variable in azuredeploy.json)
- Naming conventions (variables in azuredeploy.json)
- OpenShift cluster specifics, modified via hosts file (deployOpenShift.sh)

The [OKD template](#) has multiple branches available for different versions of OKD. Based on your needs, you can deploy directly from the repo or you can fork the repo and make custom changes before deploying.

Use the `appId` value from the service principal that you created earlier for the `aadClientId` parameter.

The following is an example of a parameters file named `azuredeploy.parameters.json` with all the required inputs.

```
{
  "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentParameters.json#",
  "contentVersion": "1.0.0.0",
  "parameters": {
    "masterVmSize": {
      "value": "Standard_E2s_v3"
    },
    "infraVmSize": {
      "value": "Standard_E2s_v3"
    },
    "nodeVmSize": {
      "value": "Standard_E2s_v3"
    },
    "storageKind": {
      "value": "managed"
    },
    "openshiftClusterPrefix": {
      "value": "mycluster"
    },
    "masterInstanceCount": {
      "value": 3
    },
    "infraInstanceCount": {
      "value": 2
    },
    "nodeInstanceCount": {
      "value": 2
    }
  }
}
```

```
"dataDiskSize": {
    "value": 128
},
"adminUsername": {
    "value": "clusteradmin"
},
"openshiftPassword": {
    "value": "{Strong Password}"
},
"sshPublicKey": {
    "value": "{SSH Public Key}"
},
"enableMetrics": {
    "value": "true"
},
"enableLogging": {
    "value": "false"
},
"keyVaultResourceGroup": {
    "value": "keyvaultrg"
},
"keyVaultName": {
    "value": "keyvault"
},
"keyVaultSecret": {
    "value": "keysecret"
},
"enableAzure": {
    "value": "true"
},
"aadClientId": {
    "value": "11111111-abcd-1234-efgh-111111111111"
},
"aadClientSecret": {
    "value": "{Strong Password}"
},
"defaultSubDomainType": {
    "value": "nipro"
}
}
```

Replace the parameters with your specific information.

Different releases may have different parameters so please verify the necessary parameters for the branch you use.

## Deploy using Azure CLI

### NOTE

The following command requires Azure CLI 2.0.8 or later. You can verify the CLI version with the `az --version` command. To update the CLI version, see [Install Azure CLI](#).

The following example deploys the OKD cluster and all related resources into a resource group named openshiftrg, with a deployment name of myOpenShiftCluster. The template is referenced directly from the GitHub repo while using a local parameters file named azuredeploy.parameters.json.

```
az deployment group create -g openshiftrg --name myOpenShiftCluster \
    --template-uri https://raw.githubusercontent.com/Microsoft/openshift-origin/master/azuredeploy.json \
    --parameters ./azuredeploy.parameters.json
```

The deployment takes at least 30 minutes to finish, based on the total number of nodes deployed. The URL of

the OpenShift console and the DNS name of the OpenShift master prints to the terminal when the deployment finishes. Alternatively, you can view the outputs section of the deployment from the Azure portal.

```
{  
  "OpenShift Console Url": "http://openshiftlb.cloudapp.azure.com/console",  
  "OpenShift Master SSH": "ssh -p 2200 clusteradmin@myopenshiftmaster.cloudapp.azure.com"  
}
```

If you don't want to tie up the command line waiting for the deployment to complete, add `--no-wait` as one of the options for the group deployment. The output from the deployment can be retrieved from the Azure portal in the deployment section for the resource group.

## Connect to the OKD cluster

When the deployment finishes, connect to the OpenShift console with your browser using the `OpenShift Console Url`. Alternatively, you can SSH to the OKD master. Following is an example that uses the output from the deployment:

```
$ ssh -p 2200 clusteradmin@myopenshiftmaster.cloudapp.azure.com
```

## Clean up resources

Use the `az group delete` command to remove the resource group, OpenShift cluster, and all related resources when they're no longer needed.

```
az group delete --name openshiftrg
```

## Next steps

- [Post-deployment tasks](#)
- [Troubleshoot OpenShift deployment](#)
- [Getting started with OKD](#)

# Use Azure to host and run SAP workload scenarios

9/21/2022 • 28 minutes to read • [Edit Online](#)

When you use Microsoft Azure, you can reliably run your mission-critical SAP workloads and scenarios on a scalable, compliant, and enterprise-proven platform. You get the scalability, flexibility, and cost savings of Azure. With the expanded partnership between Microsoft and SAP, you can run SAP applications across development and test and production scenarios in Azure and be fully supported. From SAP NetWeaver to SAP S/4HANA, SAP BI on Linux to Windows, and SAP HANA to SQL Server, Oracle, Db2, etc, we've got you covered.

Besides hosting SAP NetWeaver and S/4HANA scenarios with the different DBMS on Azure, you can host other SAP workload scenarios, like SAP BI on Azure.

We just announced our new services of Azure Center for SAP solutions and Azure Monitor for SAP 2.0 entering the public preview stage. These services will give you the possibility to deploy SAP workload on Azure in a highly automated manner in an optimal architecture and configuration. And monitor your Azure infrastructure, OS, DBMS, and ABAP stack deployments on one single pane of glass.

For customers and partners who are focussed on deploying and operating their assets in public cloud through Terraform and Ansible, leverage our SAP Deployment Automation Framework (SDAF) to jump start your SAP deployments into Azure using our public Terraform and Ansible modules on [github](#).

Hosting SAP workload scenarios in Azure also can create requirements of identity integration and single sign-on. This situation can occur when you use Azure Active Directory (Azure AD) to connect different SAP components and SAP software-as-a-service (SaaS) or platform-as-a-service (PaaS) offers. A list of such integration and single sign-on scenarios with Azure AD and SAP entities is described and documented in the section "Azure AD SAP identity integration and single sign-on."

## Changes to the SAP workload section

Changes to documents in the SAP on Azure workload section are listed at the [end of this article](#). The entries in the change log are kept for around 180 days.

## You want to know

If you have specific questions, we are going to point you to specific documents or flows in this section of the start page. You want to know:

- What Azure VMs and HANA Large Instance units are supported for which SAP software releases and which operating system versions. Read the document [What SAP software is supported for Azure deployment](#) for answers and the process to find the information
- What SAP deployment scenarios are supported with Azure VMs and HANA Large Instances. Information about the supported scenarios can be found in the documents:
  - [SAP workload on Azure virtual machine supported scenarios](#)
  - [Supported scenarios for HANA Large Instance](#)
- What Azure Services, Azure VM types and Azure storage services are available in the different Azure regions, check the site [Products available by region](#)
- Are third-party HA frameworks, besides Windows and Pacemaker supported? Check bottom part of [SAP support note #1928533](#)
- What Azure storage is best for my scenario? Read [Azure Storage types for SAP workload](#)
- Is the Red Hat kernel in Oracle Enterprise Linux supported by SAP? Read SAP [SAP support note #1565179](#)

- Why are the Azure [Da\(s\)v4/Ea\(s\)](#) VM families not certified for SAP HANA? The Azure Das/Eas VM families are based on AMD processor-driven hardware. SAP HANA does not support AMD processors, not even in virtualized scenarios
- Why am I still getting the message: 'The cpu flags for the RDTSCP instruction or the cpu flags for constant\_tsc or nonstop\_tsc are not set or current\_clocksource and available\_clocksource are not correctly configured' with SAP HANA, despite the fact that I am running the most recent Linux kernels. For the answer, check [SAP support note #2791572](#)
- Where can I find architectures for deploying SAP Fiori on Azure? Check out the blog [SAP on Azure: Application Gateway Web Application Firewall \(WAF\) v2 Setup for Internet facing SAP Fiori Apps](#)

## Documentation space

In the SAP workload documentation space, you can find the following areas:

- **SAP on Azure Large Instances:** This documentation section is covering a bare-metal service that originally was named HANA Large Instances. Different topics around this technology are covered in this section
- **Plan and Deploy (Azure VMs):** Deploying SAP workload into Azure Infrastructure as a Service, you should go through the documents in this section first to learn more about the principle Azure components used and guidelines
- **Storage (Azure VMs):** This section includes documents that give recommendations how to use the different Azure storage types when deploying SAP workload on Azure
- **DBMS Guides (Azure VMs):** The section DBMS Guides covers specifics around deploying different DBMS that are supported for SAP workload in Azure IaaS
- **High Availability (Azure VMs):** In this section, many of the high availability configurations around SAP workload on Azure is covered. This section includes detailed documentation around deploying Windows clustering and Pacemaker cluster configuration for the different SAP components and different database systems
- **Automation Framework (Azure VMs):** Automation Framework documentation is covering an a [Terraform and Ansible based automation framework](#) that allows automation of Azure infrastructure and SAP software
- **Azure Monitor for SAP solutions:** Microsoft developed monitoring solutions specifically for SAP supported OS and DBMS, as well as S/4HANA and NetWeaver. This section documents the deployment and usage of the service
- **Integration with Microsoft Services and References** contain different links to integration between SAP and other Microsoft services. The list may not be complete.

## Change Log

- September 14, 2022 Release of updated SAP on Oracle guide with new and updated content [Azure Virtual Machines Oracle DBMS deployment for SAP workload](#)
- September 8, 2022: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES](#) to add instructions for deploying /hana/shared (only) on NFS on Azure Files
- September 6, 2022: Add managed identity for pacemaker fence agent [Set up Pacemaker on SUSE Linux Enterprise Server \(SLES\) in Azure](#) on SLES and [Setting up Pacemaker on RHEL in Azure](#) RHEL
- August 22, 2022: Release of cost optimization scenario [Deploy PAS and AAS with SAP NetWeaver HA cluster on RHEL](#)
- August 09, 2022: Release of scenario [HA for SAP ASCS/ERS with NFS simple mount](#) on SLES 15 for SAP Applications
- July 18, 2022: Clarify statement around Pacemaker support on Oracle Linux in [Azure Virtual Machines Oracle DBMS deployment for SAP workload](#)
- June 29, 2022: Add recommendation and links to Pacemaker usage for Db2 versions 11.5.6 and higher in the documents [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#), [High availability of IBM](#)

[Db2 LUW on Azure VMs on SUSE Linux Enterprise Server with Pacemaker](#), and [High availability of IBM Db2 LUW on Azure VMs on Red Hat Enterprise Linux Server](#)

- June 08, 2022: Change in [HA for SAP NW on Azure VMs on SLES with ANF](#) and [HA for SAP NW on Azure VMs on RHEL with ANF](#) to adjust timeouts when using NFSv4.1 (related to NFSv4.1 lease renewal) for more resilient Pacemaker configuration
- June 02, 2022: Change in the [SAP Deployment Guide](#) to add a link to RHEL in-place upgrade documentation
- June 02, 2022: Change in [HA for SAP NetWeaver on Azure VMs on Windows with Azure NetApp Files\(SMB\)](#), [HA for SAP NW on Azure VMs on SLES with ANF](#) and [HA for SAP NW on Azure VMs on RHEL with ANF](#) to add sizing considerations
- May 11, 2022: Change in [Cluster an SAP ASCS/SCS instance on a Windows failover cluster by using a cluster shared disk in Azure](#), [Prepare the Azure infrastructure for SAP HA by using a Windows failover cluster and shared disk for SAP ASCS/SCS](#) and [SAP ASCS/SCS instance multi-SID high availability with Windows server failover clustering and Azure shared disk](#) to update instruction about the usage of Azure shared disk for SAP deployment with PPG.
- May 10, 2022: Change in [HA for SAP HANA scale-up with ANF on RHEL](#), [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#), [HA for SAP HANA Scale-up with Azure NetApp Files on SLES](#), [SAP HANA scale-out with standby node on Azure VMs with ANF on SLES](#), [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES](#) and [SAP HANA scale-out with standby node on Azure VMs with ANF on RHEL](#) to adjust parameters per SAP note 3024346
- April 26, 2022: Changes in [Setting up Pacemaker on SUSE Linux Enterprise Server in Azure](#) to add Azure Identity Python module to installation instructions for Azure Fence Agent
- March 30, 2022: Adding information that Red Hat Gluster Storage is being phased out [GlusterFS on Azure VMs on RHEL](#)
- March 30, 2022: Correcting DNN support for older releases of SQL Server in [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#)
- March 28, 2022: Formatting changes and reorganizing ILB configuration instructions in: [HA for SAP HANA on Azure VMs on SLES](#), [HA for SAP HANA Scale-up with Azure NetApp Files on SLES](#), [HA for SAP HANA on Azure VMs on RHEL](#), [HA for SAP HANA scale-up with ANF on RHEL](#), [HA for SAP NW on SLES with NFS on Azure Files](#), [HA for SAP NW on Azure VMs on SLES with ANF](#), [HA for SAP NW on Azure VMs on SLES for SAP applications](#), [HA for NFS on Azure VMs on SLES](#), [HA for SAP NNW on Azure VMs on SLES multi-SID guide](#), [HA for SAP NW on RHEL with NFS on Azure Files](#), [HA for SAP NW on Azure VMs on RHEL with ANF](#), [HA for SAP NW on Azure VMs on RHEL for SAP applications](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#)
- March 15, 2022: Corrected rsize and wsize mount option settings for ANF in [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#)
- March 1, 2022: Corrected note about database snapshots with multiple database containers in [SAP HANA Large Instances high availability and disaster recovery on Azure](#)
- February 28, 2022: Added E(d)sv5 VM storage configurations to [SAP HANA Azure virtual machine storage configurations](#)
- February 13, 2022: Corrected broken links to HANA hardware directory in the following documents: SAP Business One on Azure Virtual Machines, Available SKUs for HANA Large Instances, Certification of SAP HANA on Azure (Large Instances), Installation of SAP HANA on Azure virtual machines, SAP workload planning and deployment checklist, SAP HANA infrastructure configurations and operations on Azure, SAP HANA on Azure Large Instance migration to Azure Virtual Machines, Install and configure SAP HANA (Large Instances) ,on Azure, High availability of SAP HANA scale-out system on Red Hat Enterprise Linux, High availability for SAP HANA scale-out system with HSR on SUSE Linux Enterprise Server, High availability of SAP HANA on Azure VMs on SUSE Linux Enterprise Server, Deploy a SAP HANA scale-out system with standby node on Azure VMs by using Azure NetApp Files on SUSE Linux Enterprise Server, SAP workload on Azure virtual machine supported scenarios, What SAP software is supported for Azure deployments
- February 13, 2022: Change in [HA for SAP NetWeaver on Azure VMs on Windows with Azure NetApp](#)

[Files\(SMB\)](#) to add instructions about adding the SAP installation user as [Administrators Privilege user](#) to avoid SWPM permission errors

- February 09, 2022: Add more information around 4K sectors usage of Db2 11.5 in [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#)
- February 08, 2022: Style changes in [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#)
- February 07, 2022: Adding new functionality [ANF application volume groups for HANA](#) in documents [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#) and [Azure proximity placement groups for optimal network latency with SAP applications](#)
- January 30, 2022: Adding context about SQL Server proportional fill and expectations that SQL Server data files should be the same size and should have the same free space in [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#)
- January 24, 2022: Change in [HA for SAP NW on SLES with NFS on Azure Files](#), [HA for SAP NW on Azure VMs on SLES with ANF](#), [HA for SAP NW on Azure VMs on SLES for SAP applications](#), [HA for NFS on Azure VMs on SLES](#), [HA for SAP NNW on Azure VMs on SLES multi-SID guide](#), [HA for SAP NW on RHEL with NFS on Azure Files](#), [HA for SAP NW on Azure VMs on RHEL for SAP applications](#) and [HA for SAP NW on Azure VMs on RHEL with ANF](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#) to remove cidr\_netmask from Pacemaker configuration to allow the resource agent to determine the value automatically.
- January 12, 2022: Change in [HA for SAP NetWeaver on Azure VMs on Windows with Azure NetApp Files\(SMB\)](#) to remove obsolete information for the SAP kernel that supports the scenario.
- December 08, 2021: Change in [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#) to clarify Azure Load Balancer settings.
- December 08, 2021: Release of scenario [HA of SAP HANA Scale-up with Azure NetApp Files on SLES](#)
- December 07, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) to clarify that the instructions are applicable for both RHEL 7 and RHEL 8
- December 07, 2021: Change in [HA for SAP NW on SLES with NFS on Azure Files](#), [HA for SAP NW on Azure VMs on SLES with ANF](#) and [HA for SAP NW on Azure VMs on SLES for SAP applications](#) to adjust the instructions for configuring SWAP file.
- December 02, 2021: Introduction of new fencing method in [Setting up Pacemaker on SUSE Linux Enterprise Server in Azure](#) using Azure shared disk SBD device
- December 01, 2021: Change in [SAP ASCS/SCS instance with WSFC and file share](#), [HA for SAP NetWeaver on Azure VMs on Windows with Azure NetApp Files\(SMB\)](#) and [HA for SAP NetWeaver on Azure VMs on Windows with Azure Files\(SMB\)](#) to update the SAP kernel version, required to support clustering SAP on Windows with file share
- November 30, 2021: Added [Using Windows DFS-N to support flexible SAPMNT share creation for SMB-based file share](#)
- November 22, 2021: Change in [HA for SAP NW on SLES with NFS on Azure Files](#) and [HA for SAP NW on RHEL with NFS on Azure Files](#) to clarify the guidelines for J2EE SAP systems and share consolidations per storage account.
- November 16, 2021: Release of high availability guides for SAP ASCS/ERS with NFS on Azure files [HA for SAP NW on SLES with NFS on Azure Files](#) and [HA for SAP NW on RHEL with NFS on Azure Files](#)
- November 15, 2021: Introduction of new proximity placement architecture for zonal deployments in [Azure proximity placement groups for optimal network latency with SAP applications](#)
- November 02, 2021: Changed [Azure Storage types for SAP workload](#) and [SAP ASE Azure Virtual Machines DBMS deployment for SAP workload](#) to declare SAP ASE support for NFS on Azure NetApp Files.
- November 02, 2021: Changed [SAP workload configurations with Azure Availability Zones](#) to move Singapore SouthEast to regions for active/active configurations
- November 02, 2021: Change in [High availability of SAP HANA on Azure VMs on Red Hat Enterprise Linux](#) to update instructions for HANA scale-up Active/Active (Read Enabled) configuration.

- October 26, 2021: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) to update resource names in HANA scale-out Active/Active (Read Enabled) configuration
- October 19, 2021: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) to add instructions for HANA scale-out Active/Active (Read Enabled) configuration
- October 11, 2021: Change in [Cluster an SAP ASCS/SCS instance on a Windows failover cluster by using a cluster shared disk in Azure](#), [Prepare the Azure infrastructure for SAP HA by using a Windows failover cluster and shared disk for SAP ASCS/SCS](#) and [SAP ASCS/SCS instance multi-SID high availability with Windows server failover clustering and Azure shared disk](#) to add instructions about zone redundant storage (ZRS) for Azure shared disk support
- October 08, 2021: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES](#), [HA for SAP HANA scale-up with ANF on RHEL](#) and [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) to add defaults in sudoers file and update for HANA scale-out(for HANA srHook)
- October 01, 2021: Added link to new Azure Backup architecture for SAP HANA backup document into table of content. Added link to Azure Backup service for Oracle DBMS into [Azure Virtual Machines Oracle DBMS deployment for SAP workload](#)
- September 24, 2021: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES](#), [HA for SAP HANA scale-up with ANF on RHEL](#) and [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) to adjust the instructions for creating sudoers file (for HANA srHook)
- September 16, 2021: Release of [HA for SAP NetWeaver on Azure VMs on Windows with Azure Files\(SMB\)](#)
- September 15, 2021: Introducing new HADR configuration for SAP ASE in [SAP ASE Azure Virtual Machines DBMS deployment for SAP workload](#)
- September 08, 2021: Adding manual QoS capacity pool into [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#)
- August 26, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) and [Setting up Pacemaker on SLES in Azure](#) to correct the role definition JSON for Azure Fence Agent
- August 17, 2021: Changes in [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#), [Azure Storage types for SAP workload](#), and [SAP workload on Azure virtual machine supported scenarios](#) to introduce support for IBM Db2 using NFS volumes hosted on ANF
- August 02, 2021: Change in [HA for SAP NW on Azure VMs on SLES for SAP applications](#), [HA for SAP NW on Azure VMs on SLES with ANF](#), [HA for SAP NW on Azure VMs on RHEL for SAP applications](#) and [HA for SAP NW on Azure VMs on RHEL with ANF](#) to clarify the behavior (ENSA1/ENSA2) for a test scenario, simulating enqueue server failure
- August 11, 2021: Change in [HA for SAP NW on Azure VMs on RHEL for SAP applications](#), [HA for SAP NW on Azure VMs on RHEL with ANF](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#) to adjust cluster resources stickiness, migration thresholds and order constraints
- August 11, 2021: Release of [SAP BW-Near Line Storage \(NLS\) implementation guide with SAP IQ on Azure](#)
- July 29, 2021: Introduce combined two-node Windows cluster for ASCS/SCS and DBMS in [High availability for SAP NetWeaver on Azure VMs on Windows with Azure NetApp Files\(SMB\) for SAP applications](#) and [Cluster an SAP ASCS/SCS instance on a Windows failover cluster by using a cluster shared disk in Azure](#)
- July 26, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) and [Setting up Pacemaker on SLES in Azure](#) to replace role assignment instructions with links to the RBAC documentation in the sections describing the set up for Azure Fence Agent
- July 22, 2021: Change in [HA for SAP NW on Azure VMs on RHEL for SAP applications](#), [HA for SAP NW on Azure VMs on RHEL with ANF](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#) to remove `failure-timeout` for the ASCS cluster resource (ENSA2 only)
- July 16, 2021: Restructuring of the SAP on Azure documentation Table of contents(TOC) for more streamlined navigation
- July 2, 2021: Change in [Backup and restore of SAP HANA on HANA Large Instances](#) to remove duplicate content for azacsnap tool and backup and restore of HANA Large Instances
- July 2, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) to add information how to avoid fence race

in two node Pacemaker cluster and a link to KB, explaining how to reduce failover delays when using optional fencing configuration with `fence_kdump`

- July 1, 2021: Adding new certified HANA Large Instances SKUs in [Available SKUs for HLI](#)
- June 30, 2021: Change in [HA guide for SAP ASCS/SCS with WSFC and Azure NetApp Files\(SMB\)](#) to add a section for recommended SAP profile parameters
- June 29, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) to add optional fencing configuration with `fence_kdump`
- June 28, 2021: Change in [HA guide for SAP ASCS/SCS with WSFC and Azure NetApp Files\(SMB\)](#) to add a statement that the SMB Server (Computer Account) Prefix should be no longer than 8 characters to avoid running into SAP hostname length limitation
- June 17, 2020: Change in [High availability of SAP HANA on Azure VMs on RHEL](#) to remove meta keyword from HANA resource creation command (RHEL 8.x)
- June 09, 2021: Correct VM SKU names for M192---\_v2 in [SAP HANA Azure virtual machine storage configurations](#)
- May 26, 2021: Change in [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES, HA for SAP HANA scale-up with ANF on RHEL](#) and [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) to add configuration to prepare the OS for running HANA on ANF
- May 13, 2021: Change in [Setting up Pacemaker on SLES in Azure](#) to clarify how resource agent `azure-events` operates
- April 30, 2021: Change in [Setting up Pacemaker on SLES in Azure](#) to include warning about incompatible change with Azure Fence Agent in a version of package `python3-azure-mgmt-compute` (SLES 15)
- April 27, 2021: Change in [SAP ASCS/SCS instance with WSFC and file share](#) to add links to important SAP notes in the prerequisites section
- April 27, 2021: Added new Msv2, Mdsv2 VMs into HANA storage configuration in [SAP HANA Azure virtual machine storage configurations](#)
- April 27, 2021: Added requirement for using same storage types in HANA System Replication across all VMs of HSR configuration in [SAP HANA Azure virtual machine storage configurations](#)
- April 27, 2021: Added requirement for using same storage types in DBMS replication scenarios across all VMs of DBMS high availability replication configurations in [Azure Storage types for SAP workload](#)
- April 23, 2021: Added section to configure private link for Azure database for MySQL and some minor changes in [SAP BusinessObjects BI platform deployment guide for linux on Azure](#)
- April 22, 2021: Release of SAP BusinessObjects BI Platform for Windows on Azure documentation, [SAP BusinessObjects BI platform deployment guide for Windows on Azure](#)
- April 21, 2021: Add explanation why HCMT/HWCCT storage tests on M32ts and M32ls might fall short of HANA KPIs when enabling read cache for the Premium storage disks in article [SAP HANA Azure virtual machine storage configurations](#)
- April 20, 2021: Clarify storage block sizes for IBM Db2 with different Azure block storage in article [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#)
- April 12, 2021: Change in [HA for SAP HANA on Azure VMs on SLES, HA for SAP HANA on Azure VMs on RHEL](#) and [HA for SAP HANA scale-up with ANF on RHEL](#) to add configuration instructions for SAP HANA system replication Python hook
- April 12, 2021: Replaced backup documentation for SAP HANA by documents of [SAP HANA backup/restore with Azure Backup service](#)
- April 12, 2021: Release of [SAP HANA scale-out HSR with Pacemaker on Azure VMs on SLES](#) configuration guide
- April 07, 2021: Clarified support for SQL Server multi-instance and multi-database support in [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#)
- April 07, 2021: Added information related to secondary IP addresses in [Azure Virtual Machines planning and implementation for SAP NetWeaver](#)

- April 07, 2021: added support for Oracle DBMS support on ANF in [Azure Storage types for SAP workload](#)
- March 17, 2021: Change in [HA for SAP HANA on Azure VMs on SLES](#), [HA for SAP HANA on Azure VMs on RHEL](#) and [HA for SAP HANA scale-up with ANF on RHEL](#) to add instructions for HANA Active/Read-enabled system replication in Pacemaker cluster
- March 15, 2021: Change in [SAP ASCS/SCS instance with WSFC and file share](#),[Install SAP ASCS/SCS instance with WSFC and file share](#) and [SAP ASCS/SCS multi-SID with WSFC and file share](#) to clarify that the SAP ASCS/SCS instances and the SOFS share must be deployed in separate clusters
- March 03, 2021: Change in [HA guide for SAP ASCS/SCS with WSFC and Azure NetApp Files\(SMB\)](#) to add a cautionary statement that elevated privileges are required for the user running SWPM, during the installation of the SAP system
- February 11, 2021: Changes in [High availability of IBM Db2 LUW on Azure VMs on Red Hat Enterprise Linux Server](#) to amend pacemaker cluster commands for RHEL 8.x
- February 03, 2021: Change in [Setting up Pacemaker on RHEL in Azure](#) to update pcmk\_host\_map in the `stonith create` command
- February 03, 2021: Change in [Setting up Pacemaker on SLES in Azure](#) to add pcmk\_host\_map in the `stonith create` command
- February 03, 2021: More details on I/O scheduler settings for SUSE in article [SAP HANA Azure virtual machine storage configurations](#)
- February 01, 2021: Change in [HA for SAP HANA scale-up with ANF on RHEL](#), [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#), [SAP HANA scale-out with standby node on Azure VMs with ANF on SLES](#) and [SAP HANA scale-out with standby node on Azure VMs with ANF on RHEL](#) to add a link to [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#)
- January 23, 2021: Introduce the functionality of HANA data volume partitioning as functionality to stripe I/O operations against HANA data files across different Azure disks or NFS shares without using a disk volume manager in articles [SAP HANA Azure virtual machine storage configurations](#) and [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#)
- January 18, 2021: Added support of Azure net Apps Files based NFS for Oracle in [Azure Virtual Machines Oracle DBMS deployment for SAP workload](#) and adjusting decimals in table in document [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#)
- January 11, 2021: Minor changes in [HA for SAP NW on Azure VMs on RHEL for SAP applications](#), [HA for SAP NW on Azure VMs on RHEL with ANF](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#) to adjust commands to work for both RHEL8 and RHEL7, and ENSA1 and ENSA2
- January 05, 2021: Changes in [SAP HANA scale-out with standby node on Azure VMs with ANF on SLES](#) and [SAP HANA scale-out with standby node on Azure VMs with ANF on RHEL](#), revising the recommended configuration to allow SAP Host Agent to manage the local port range
- January 04, 2021: Add new Azure regions supported by HLI into [What is SAP HANA on Azure \(Large Instances\)](#)
- December 29, 2020: Add architecture recommendations for specific Azure regions in [SAP workload configurations with Azure Availability Zones](#)
- December 21, 2020: Add new certifications to SKUs of HANA Large Instances in [Available SKUs for HLI](#)
- December 12, 2020: Added pointer to SAP note clarifying details on Oracle Enterprise Linux support by SAP to [What SAP software is supported for Azure deployments](#)
- November 26, 2020: Adapt [SAP HANA Azure virtual machine storage configurations](#) and [Azure Storage types for SAP workload](#) to changed single VM SLAs
- November 05, 2020: Changing link to new SAP note about HANA supported file system types in [SAP HANA Azure virtual machine storage configurations](#)
- October 26, 2020: Changing some tables for Azure premium storage configuration to clarify provisioned versus burst throughput in [SAP HANA Azure virtual machine storage configurations](#)
- October 22, 2020: Change in [HA for SAP NW on Azure VMs on SLES for SAP applications](#), [HA for SAP NW on](#)

Azure VMs on SLES with ANF, HA for SAP NW on Azure VMs on RHEL for SAP applications and HA for SAP NW on Azure VMs on RHEL with ANF to adjust the recommendation for net.ipv4.tcp\_keepalive\_time

- October 16, 2020: Change in HA of IBM Db2 LUW on Azure VMs on SLES with Pacemaker, HA for SAP NW on Azure VMs on RHEL for SAP applications, HA of IBM Db2 LUW on Azure VMs on RHEL, HA for SAP NW on Azure VMs on RHEL multi-SID guide, HA for SAP NW on Azure VMs on RHEL with ANF, HA for SAP NW on Azure VMs on SLES for SAP applications, HA for SAP NNW on Azure VMs on SLES multi-SID guide, HA for SAP NW on Azure VMs on SLES with ANF for SAP applications, HA for NFS on Azure VMs on SLES, HA of SAP HANA on Azure VMs on SLES, HA for SAP HANA scale-up with ANF on RHEL, HA of SAP HANA on Azure VMs on RHEL, SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL, Prepare Azure infrastructure for SAP ASCS/SCS with WSFC and shared disk, multi-SID HA guide for SAP ASCS/SCS with WSFC and Azure shared disk and multi-SID HA guide for SAP ASCS/SCS with WSFC and shared disk to add a statement that floating IP is not supported in load-balancing scenarios on secondary IPs
- October 16, 2020: Adding documentation to control storage snapshots of HANA Large Instances in [Backup and restore of SAP HANA on HANA Large Instances](#)
- October 15, 2020: Release of SAP BusinessObjects BI Platform on Azure documentation, [SAP BusinessObjects BI platform planning and implementation guide on Azure](#) and [SAP BusinessObjects BI platform deployment guide for linux on Azure](#)
- October 05, 2020: Release of [SAP HANA scale-out HSR with Pacemaker on Azure VMs on RHEL](#) configuration guide
- September 30, 2020: Change in [High availability of SAP HANA on Azure VMs on RHEL](#), [HA for SAP HANA scale-up with ANF on RHEL](#) and [Setting up Pacemaker on RHEL in Azure](#) to adapt the instructions for RHEL 8.1
- September 29, 2020: Making restrictions and recommendations around usage of PPG more obvious in the article [Azure proximity placement groups for optimal network latency with SAP applications](#)
- September 28, 2020: Adding a new storage operation guide for SAP HANA using Azure NetApp Files with the document [NFS v4.1 volumes on Azure NetApp Files for SAP HANA](#)
- September 23, 2020: Add new certified SKUs for HLI in [Available SKUs for HLI](#)
- September 20, 2020: Changes in documents [Considerations for Azure Virtual Machines DBMS deployment for SAP workload](#), [SQL Server Azure Virtual Machines DBMS deployment for SAP NetWeaver](#), [Azure Virtual Machines Oracle DBMS deployment for SAP workload](#), [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#) to adapt to new configuration suggestion that recommends separation of DBMS binaries and SAP binaries into different Azure disks. Also adding Ultra disk recommendations to the different guides.
- September 08, 2020: Change in [High availability of SAP HANA on Azure VMs on SLES](#) to clarify fencing definitions
- September 03, 2020: Change in [SAP HANA Azure virtual machine storage configurations](#) to adapt to minimal 2 IOPS per 1 GB capacity with Ultra disk
- September 02, 2020: Change in [Available SKUs for HLI](#) to get more transparent in what SKUs are HANA certified
- August 25, 2020: Change in [HA for SAP NW on Azure VMs on SLES with ANF](#) to fix typo
- August 25, 2020: Change in [HA guide for SAP ASCS/SCS with WSFC and shared disk](#), [Prepare Azure infrastructure for SAP ASCS/SCS with WSFC and shared disk](#) and [Install SAP NW HA with WSFC and shared disk](#) to introduce the option of using Azure shared disk and document SAP ERS2 architecture
- August 25, 2020: Release of [multi-SID HA guide for SAP ASCS/SCS with WSFC and Azure shared disk](#)
- August 25, 2020: Change in [HA guide for SAP ASCS/SCS with WSFC and Azure NetApp Files\(SMB\)](#), [Prepare Azure infrastructure for SAP ASCS/SCS with WSFC and file share](#), [multi-SID HA guide for SAP ASCS/SCS with WSFC and shared disk](#) and [multi-SID HA guide for SAP ASCS/SCS with WSFC and SOFS file share](#) as a result of the content updates and restructuring in the HA guides for SAP ASCS/SCS with WFC and shared disk
- August 21, 2020: Adding new OS release into [Compatible Operating Systems for HANA Large Instances](#) as available operating system for HLI units of type I and II

- August 18, 2020: Release of [HA for SAP HANA scale-up with ANF on RHEL](#)
- August 17, 2020: Add information about using Azure Site Recovery for moving SAP NetWeaver systems from on-premises to Azure in article [Azure Virtual Machines planning and implementation for SAP NetWeaver](#)
- 08/14/2020: Adding disk configuration advice for Db2 in article [IBM Db2 Azure Virtual Machines DBMS deployment for SAP workload](#)
- August 11, 2020: Adding RHEL 7.6 into [Compatible Operating Systems for HANA Large Instances](#) as available operating system for HLI units of type I
- August 10, 2020: Introducing cost conscious SAP HANA storage configuration in [SAP HANA Azure virtual machine storage configurations](#) and making some updates to [SAP workloads on Azure: planning and deployment checklist](#)
- August 04, 2020: Change in [Setting up Pacemaker on SLES in Azure](#) and [Setting up Pacemaker on RHEL in Azure](#) to emphasize the importance of reliable name resolution for Pacemaker clusters
- August 04, 2020: Change in [SAP NW HA on WFCs with file share](#), [SAP NW HA on WFCs with shared disk](#), [HA for SAP NW on Azure VMs](#), [HA for SAP NW on Azure VMs on SLES](#), [HA for SAP NW on Azure VMs on SLES with ANF](#), [HA for SAP NW on Azure VMs on SLES multi-SID guide](#), [High availability for SAP NetWeaver on Azure VMs on RHEL](#), [HA for SAP NW on Azure VMs on RHEL with ANF](#) and [HA for SAP NW on Azure VMs on RHEL multi-SID guide](#) to clarify the use of parameter `enqueue/encni/set_so_keepalive`
- July 23, 2020: Added the [Save on SAP HANA Large Instances with an Azure reservation](#) article explaining what you need to know before you buy an SAP HANA Large Instances reservation and how to make the purchase
- July 16, 2020: Describe how to use Azure PowerShell to install new VM Extension for SAP in the [Deployment Guide](#)
- July 04, 2020: Release of [Azure Monitor for SAP solutions \(preview\)](#)
- July 01, 2020: Suggesting less expensive storage configuration based on Azure premium storage burst functionality in document [SAP HANA Azure virtual machine storage configurations](#)
- June 24, 2020: Change in [Setting up Pacemaker on SLES in Azure](#) to release new improved Azure Fence Agent and more resilient fencing configuration for devices, based on Azure Fence Agent
- June 24, 2020: Change in [Setting up Pacemaker on RHEL in Azure](#) to release more resilient fencing configuration
- June 23, 2020: Changes to [Azure Virtual Machines planning and implementation for SAP NetWeaver](#) guide and introduction of [Azure Storage types for SAP workload](#) guide
- June 22, 2020: Add installation steps for new VM Extension for SAP to the [Deployment Guide](#)
- June 16, 2020: Change in [Public endpoint connectivity for VMs using Azure Standard ILB in SAP HA scenarios](#) to add a link to SUSE Public Cloud Infrastructure 101 documentation
- June 10, 2020: Adding new HLI SKUs into [Available SKUs for HLI](#) and [SAP HANA \(Large Instances\) storage architecture](#)

# Create an Oracle Database in an Azure VM

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This guide details using the Azure CLI to deploy an Azure virtual machine from the [Oracle marketplace gallery image](#) in order to create an Oracle 19c database. Once the server is deployed, you will connect via SSH in order to configure the Oracle database.

If you don't have an Azure subscription, create a [free account](#) before you begin.

If you choose to install and use the CLI locally, this quickstart requires that you are running the Azure CLI version 2.0.4 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create a resource group

Create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed.

The following example creates a resource group named `rg-oracle` in the `eastus` location.

```
az group create --name rg-oracle --location eastus
```

## Create virtual machine

To create a virtual machine (VM), use the `az vm create` command.

The following example creates a VM named `vmoracle19c`. It also creates SSH keys, if they do not already exist in a default key location. To use a specific set of keys, use the `--ssh-key-value` option.

```
az vm create ^
--resource-group rg-oracle ^
--name vmoracle19c ^
--image Oracle:oracle-database-19-3:oracle-database-19-0904:latest ^
--size Standard_DS2_v2 ^
--admin-username azureuser ^
--generate-ssh-keys ^
--public-ip-address-allocation static ^
--public-ip-address-dns-name vmoracle19c
```

After you create the VM, Azure CLI displays information similar to the following example. Note the value for `publicIpAddress`. You use this address to access the VM.

```
{  
  "fqdns": "",  
  "id": "/subscriptions/{snip}/resourceGroups/rg-oracle/providers/Microsoft.Compute/virtualMachines/vmoracle19c",  
  "location": "eastus",  
  "macAddress": "00-0D-3A-36-2F-56",  
  "powerState": "VM running",  
  "privateIpAddress": "10.0.0.4",  
  "publicIpAddress": "13.64.104.241",  
  "resourceGroup": "rg-oracle"  
}
```

## Create and attach a new disk for Oracle datafiles and FRA

```
az vm disk attach --name oradata01 --new --resource-group rg-oracle --size-gb 64 --sku StandardSSD_LRS --vm-name vmoracle19c
```

## Open ports for connectivity

In this task you must configure some external endpoints for the database listener to use by setting up the Azure Network Security Group that protects the VM.

1. To open the endpoint that you use to access the Oracle database remotely, create a Network Security Group rule as follows:

```
az network nsg rule create ^  
  --resource-group rg-oracle ^  
  --nsg-name vmoracle19cNSG ^  
  --name allow-oracle ^  
  --protocol tcp ^  
  --priority 1001 ^  
  --destination-port-range 1521
```

2. To open the endpoint that you use to access Oracle remotely, create a Network Security Group rule with az network nsg rule create as follows:

```
az network nsg rule create ^  
  --resource-group rg-oracle ^  
  --nsg-name vmoracle19cNSG ^  
  --name allow-oracle-EM ^  
  --protocol tcp ^  
  --priority 1002 ^  
  --destination-port-range 5502
```

3. If needed, obtain the public IP address of your VM again with az network public-ip show as follows:

```
az network public-ip show ^  
  --resource-group rg-oracle ^  
  --name vmoracle19cPublicIP ^  
  --query "ipAddress" ^  
  --output tsv
```

## Prepare the VM environment

1. Connect to the VM

To create an SSH session with the VM, use the following command. Replace the IP address with the `<publicIpAddress>` value for your VM.

```
ssh azureuser@<publicIpAddress>
```

## 2. Switch to the root user

```
sudo su -
```

## 3. Check for last created disk device that we will format for use holding Oracle datafiles

```
ls -alt /dev/sd*|head -1
```

The output will be similar to this:

```
brw-rw----. 1 root disk 8, 16 Dec 8 22:57 /dev/sdc
```

## 4. Format the device. As root user run parted on the device

First create a disk label:

```
parted /dev/sdc mklabel gpt
```

Then create a primary partition spanning the whole disk:

```
parted -a optimal /dev/sdc mkpart primary 0GB 64GB
```

Finally check the device details by printing its metadata:

```
parted /dev/sdc print
```

The output should look similar to this:

```
# parted /dev/sdc print
Model: Msft Virtual Disk (scsi)
Disk /dev/sdc: 68.7GB
Sector size (logical/physical): 512B/4096B
Partition Table: gpt
Disk Flags:
Number  Start   End     Size    File system  Name     Flags
 1      1049kB  64.0GB  64.0GB  ext4        primary
```

## 5. Create a filesystem on the device partition

```
mkfs -t ext4 /dev/sdc1
```

## 6. Create a mount point

```
mkdir /u02
```

7. Mount the disk

```
mount /dev/sdc1 /u02
```

8. Change permissions on the mount point

```
chmod 777 /u02
```

9. Add the mount to the /etc/fstab file.

```
echo "/dev/sdc1           /u02           ext4      defaults      0 0" >> /etc/fstab
```

10. Update the */etc/hosts* file with the public IP and hostname.

Change the *Public IP and VMname* to reflect your actual values:

```
echo "<Public IP> <VMname>.eastus.cloudapp.azure.com <VMname>" >> /etc/hosts
```

11. Update the hostname file

Use the following command to add the domain name of the VM to the */etc/hostname* file. This assumes you have created your resource group and VM in the **eastus** region:

```
sed -i 's/$/.eastus\.cloudapp\.azure\.com &/' /etc/hostname
```

12. Open firewall ports

As SELinux is enabled by default on the Marketplace image we need to open the firewall to traffic for the database listening port 1521, and Enterprise Manager Express port 5502. Run the following commands as root user:

```
firewall-cmd --zone=public --add-port=1521/tcp --permanent  
firewall-cmd --zone=public --add-port=5502/tcp --permanent  
firewall-cmd --reload
```

## Create the database

The Oracle software is already installed on the Marketplace image. Create a sample database as follows.

1. Switch to the **oracle** user:

```
sudo su - oracle
```

2. Start the database listener

```
lsnrctl start
```

The output is similar to the following:

```
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 20-OCT-2020 01:58:18

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Log messages written to /u01/app/oracle/diag/tnslsnr/vmoracle19c/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=vmoracle19c.eastus.cloudapp.azure.com)
(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version        TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date      20-OCT-2020 01:58:18
Uptime         0 days 0 hr. 0 min. 0 sec
Trace Level    off
Security        ON: Local OS Authentication
SNMP           OFF
Listener Log File   /u01/app/oracle/diag/tnslsnr/vmoracle19c/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=vmoracle19c.eastus.cloudapp.azure.com)(PORT=1521)))
The listener supports no services
The command completed successfully
```

### 3. Create a data directory for the Oracle data files:

```
mkdir /u02/oradata
```

### 4. Run the Database Creation Assistant:

```
dbca -silent \
-createDatabase \
-templateName General_Purpose.dbc \
-gdbname oratest1 \
-sid oratest1 \
-responseFile NO_VALUE \
-characterSet AL32UTF8 \
-sysPassword OraPasswd1 \
-systemPassword OraPasswd1 \
-createAsContainerDatabase false \
-databaseType MULTIPURPOSE \
-automaticMemoryManagement false \
-storageType FS \
-datafileDestination "/u02/oradata/" \
-ignorePreReqs
```

It takes a few minutes to create the database.

You will see output that looks similar to the following:

```
Prepare for db operation
10% complete
Copying database files
40% complete
Creating and starting Oracle instance
42% complete
46% complete
50% complete
54% complete
60% complete
Completing Database Creation
66% complete
69% complete
70% complete
Executing Post Configuration Actions
100% complete
Database creation complete. For details check the logfiles at:
/u01/app/oracle/cfgtoollogs/dbca/oratest1.
Database Information:
Global Database Name:oratest1
System Identifier(SID):oratest1
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/oratest1/oratest1.log" for further details.
```

## 5. Set Oracle variables

Before you connect, you need to set the environment variable *ORACLE\_SID*:

```
export ORACLE_SID=oratest1
```

You should also add the *ORACLE\_SID* variable to the `oracle` users `.bashrc` file for future sign-ins using the following command:

```
echo "export ORACLE_SID=oratest1" >> ~oracle/.bashrc
```

## Automate database startup and shutdown

The Oracle database by default doesn't automatically start when you restart the VM. To set up the Oracle database to start automatically, first sign in as root. Then, create and update some system files.

### 1. Sign on as root

```
sudo su -
```

### 2. Run the following command to change the automated startup flag from `N` to `Y` in the `/etc/oratab` file:

```
sed -i 's/:N/:Y/' /etc/oratab
```

### 3. Create a file named `/etc/init.d/dbora` and paste the following contents:

```
#!/bin/sh
# chkconfig: 345 99 10
# Description: Oracle auto start-stop script.
#
# Set ORA_HOME to be equivalent to $ORACLE_HOME.
ORA_HOME=/u01/app/oracle/product/19.0.0/dbhome_1
ORA_OWNER=oracle

case "$1" in
'start')
    # Start the Oracle databases:
    # The following command assumes that the Oracle sign-in
    # will not prompt the user for any values.
    # Remove "&" if you don't want startup as a background process.
    su - $ORA_OWNER -c "$ORA_HOME/bin/dbstart $ORA_HOME" &
    touch /var/lock/subsys/dbora
    ;;
'stop')
    # Stop the Oracle databases:
    # The following command assumes that the Oracle sign-in
    # will not prompt the user for any values.
    su - $ORA_OWNER -c "$ORA_HOME/bin/dbshut $ORA_HOME" &
    rm -f /var/lock/subsys/dbora
    ;;
esac
```

4. Change permissions on files with `chmod` as follows:

```
chgrp dba /etc/init.d/dbora
chmod 750 /etc/init.d/dbora
```

5. Create symbolic links for startup and shutdown as follows:

```
ln -s /etc/init.d/dbora /etc/rc.d/rc0.d/K01dbora
ln -s /etc/init.d/dbora /etc/rc.d/rc3.d/S99dbora
ln -s /etc/init.d/dbora /etc/rc.d/rc5.d/S99dbora
```

6. To test your changes, restart the VM:

```
reboot
```

## Clean up resources

Once you have finished exploring your first Oracle database on Azure and the VM is no longer needed, you can use the [az group delete](#) command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroup
```

## Next steps

Understand how to protect your database in Azure with [Oracle Backup Strategies](#)

Learn about other [Oracle solutions on Azure](#).

Try the [Installing and Configuring Oracle Automated Storage Management](#) tutorial.

# Install the Elastic Stack (ELK) on an Azure VM

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article walks you through how to deploy [Elasticsearch](#), [Logstash](#), and [Kibana](#), on an Ubuntu VM in Azure. To see the Elastic Stack in action, you can optionally connect to Kibana and work with some sample logging data.

In this tutorial you learn how to:

- Create an Ubuntu VM in an Azure resource group
- Install Elasticsearch, Logstash, and Kibana on the VM
- Send sample data to Elasticsearch with Logstash
- Open ports and work with data in the Kibana console

This deployment is suitable for basic development with the Elastic Stack. For more on the Elastic Stack, including recommendations for a production environment, see the [Elastic documentation](#) and the [Azure Architecture Center](#).

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This article requires version 2.0.4 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group

Create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed.

The following example creates a resource group named `myResourceGroup` in the `eastus` location.

```
az group create --name myResourceGroup --location eastus
```

# Create a virtual machine

Create a VM with the [az vm create](#) command.

The following example creates a VM named *myVM* and creates SSH keys if they do not already exist in a default key location. To use a specific set of keys, use the `--ssh-key-value` option.

```
az vm create \
    --resource-group myResourceGroup \
    --name myVM \
    --image UbuntuLTS \
    --admin-username azureuser \
    --generate-ssh-keys
```

When the VM has been created, the Azure CLI shows information similar to the following example. Take note of the `publicIpAddress`. This address is used to access the VM.

```
{
  "fqdns": "",
  "id": "/subscriptions/<subscription
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",
  "location": "eastus",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "40.68.254.142",
  "resourceGroup": "myResourceGroup"
}
```

## SSH into your VM

If you don't already know the public IP address of your VM, run the [az network public-ip list](#) command:

```
az network public-ip list --resource-group myResourceGroup --query [].ipAddress
```

Use the following command to create an SSH session with the virtual machine. Substitute the correct public IP address of your virtual machine. In this example, the IP address is *40.68.254.142*.

```
ssh azureuser@40.68.254.142
```

## Install the Elastic Stack

Import the Elasticsearch signing key and update your APT sources list to include the Elastic package repository:

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
echo "deb https://artifacts.elastic.co/packages/5.x/apt stable main" | sudo tee -a
/etc/apt/sources.list.d/elastic-5.x.list
```

Install the Java Virtual on the VM and configure the `JAVA_HOME` variable-this is necessary for the Elastic Stack components to run.

```
sudo apt update && sudo apt install openjdk-8-jre-headless
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

Run the following commands to update Ubuntu package sources and install Elasticsearch, Kibana, and Logstash.

```
sudo apt update && sudo apt install elasticsearch kibana logstash
```

**NOTE**

Detailed installation instructions, including directory layouts and initial configuration, are maintained in [Elastic's documentation](#)

## Start Elasticsearch

Start Elasticsearch on your VM with the following command:

```
sudo systemctl start elasticsearch.service
```

This command produces no output, so verify that Elasticsearch is running on the VM with this `curl` command:

```
sudo curl -XGET 'localhost:9200/'
```

If Elasticsearch is running, you see output like the following:

```
{
  "name" : "w6Z4NwR",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "SDzCajBoSK2EkXmHvJVaDQ",
  "version" : {
    "number" : "5.6.3",
    "build_hash" : "1a2f265",
    "build_date" : "2017-10-06T20:33:39.012Z",
    "build_snapshot" : false,
    "lucene_version" : "6.6.1"
  },
  "tagline" : "You Know, for Search"
}
```

## Start Logstash and add data to Elasticsearch

Start Logstash with the following command:

```
sudo systemctl start logstash.service
```

Test Logstash in interactive mode to make sure it's working correctly:

```
sudo /usr/share/logstash/bin/logstash -e 'input { stdin { } } output { stdout {} }'
```

This is a basic Logstash [pipeline](#) that echoes standard input to standard output.

```
The stdin plugin is now waiting for input:
hello azure
2017-10-11T20:01:08.904Z myVM hello azure
```

Set up Logstash to forward the kernel messages from this VM to Elasticsearch. Create a new file in an empty directory called `vm-syslog-logstash.conf` and paste in the following Logstash configuration:

```
input {
    stdin {
        type => "stdin-type"
    }

    file {
        type => "syslog"
        path => [ "/var/log/*.log", "/var/log/*/*.log", "/var/log/messages", "/var/log/syslog" ]
        start_position => "beginning"
    }
}

output {
    stdout {
        codec => rubydebug
    }
    elasticsearch {
        hosts => "localhost:9200"
    }
}
```

Test this configuration and send the syslog data to Elasticsearch:

```
sudo /usr/share/logstash/bin/logstash -f vm-syslog-logstash.conf
```

You see the syslog entries in your terminal echoed as they are sent to Elasticsearch. Use `CTRL+C` to exit out of Logstash once you've sent some data.

## Start Kibana and visualize the data in Elasticsearch

Edit `/etc/kibana/kibana.yml` and change the IP address Kibana listens on so you can access it from your web browser.

```
server.host: "0.0.0.0"
```

Start Kibana with the following command:

```
sudo systemctl start kibana.service
```

Open port 5601 from the Azure CLI to allow remote access to the Kibana console:

```
az vm open-port --port 5601 --resource-group myResourceGroup --name myVM
```

Open up the Kibana console and select **Create** to generate a default index based on the syslog data you sent to Elasticsearch earlier.

The screenshot shows the Kibana Management interface. On the left, there's a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The main area is titled 'Configure an index pattern' and contains instructions about index patterns. It has sections for 'Index pattern' (with 'logstash-\*' selected), 'Time Filter field name' (set to '@timestamp'), and checkboxes for 'Expand index pattern when searching [DEPRECATED]' and 'Use event times to create index names [DEPRECATED]'. A large blue 'Create' button at the bottom is highlighted with a red box.

Select **Discover** on the Kibana console to search, browse, and filter through the syslog events.

The screenshot shows the Kibana Discover interface. The sidebar on the left has 'Discover' selected. The main area displays a histogram with a single bar at 14:05:00, indicating a count of approximately 12. Below the histogram is a table of log entries. The table columns are Time, @timestamp, @version, message, and type. The first entry is:

Time	@timestamp	@version	message	type
October 13th 2017, 14:05:25.688	October 13th 2017, 14:05:25.688	1	[2017-10-13T21:05:24,910] [WARN ] [o.e.d.i.m.TypeParsers ] field [include_in_all] is deprecated, as [._all] is deprecated, and will be disallowed in 6.0, use [copy_to] instead.	syslog

There are two more entries in the table, both with identical details.

## Next steps

In this tutorial, you deployed the Elastic Stack into a development VM in Azure. You learned how to:

- Create an Ubuntu VM in an Azure resource group
- Install Elasticsearch, Logstash, and Kibana on the VM
- Send sample data to Elasticsearch from Logstash
- Open ports and work with data in the Kibana console

# Mainframe rehosting on Azure virtual machines

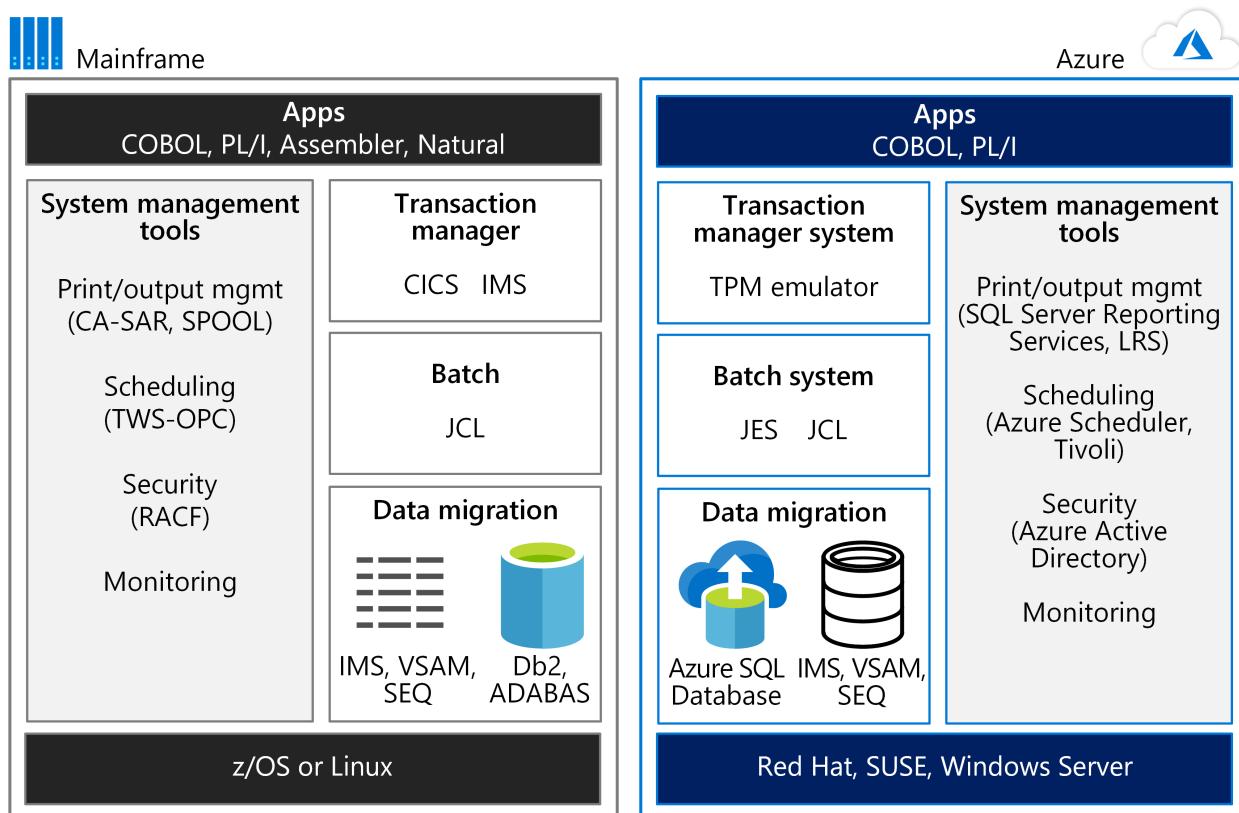
9/21/2022 • 4 minutes to read • [Edit Online](#)

Migrating workloads from mainframe environments to the cloud enables you to modernize your infrastructure and often save on costs. Many workloads can be transferred to Azure with only minor code changes, such as updating the names of databases.

Generally speaking, the term *mainframe* means a large computer system. Specifically, the vast majority currently in use are IBM System Z servers or IBM plug-compatible systems that run MVS, DOS, VSE, OS/390, or z/OS.

An Azure virtual machine (VM) is used to isolate and manage the resources for a specific application on a single instance. Mainframes such as IBM z/OS use Logical Partitions (LPARS) for this purpose. A mainframe might use one LPAR for a CICS region with associated COBOL programs, and a separate LPAR for IBM Db2 database. A typical [n-tier application on Azure](#) deploys Azure VMs into a virtual network that can be segmented into subnets for each tier.

Azure VMs can run mainframe emulation environments and compilers that support lift-and-shift scenarios. Development and testing are often among the first workloads to migrate from a mainframe to an Azure dev/test environment. Common server components that you can emulate include online transaction process (OLTP), batch, and data ingestion systems as the following figure shows.



Some mainframe workloads can be migrated to Azure with relative ease, while others can be rehosted on Azure using a partner solution. For detailed guidance about choosing a partner solution, the [Azure Mainframe Migration center](#) can help.

## Mainframe migration

Rehost, rebuild, replace, or retire? IaaS or PaaS? To determine the right migration strategy for your mainframe application, see the [Mainframe migration](#) guide in the Azure Architecture Center.

## Micro Focus rehosting platform

Micro Focus Enterprise Server is one of the largest mainframe rehosting platforms available. You can use it run your z/OS workloads on a less expensive x86 platform on Azure.

To get started:

- [Install Enterprise Server and Enterprise Developer on Azure](#)
- [Set up CICS BankDemo for Enterprise Developer on Azure](#)
- [Run Enterprise Server in a Docker Container on Azure](#)

## TmaxSoft OpenFrame on Azure

TmaxSoft OpenFrame is a popular mainframe rehosting solution used in lift-and-shift scenarios. An OpenFrame environment on Azure is suitable for development, demos, testing, or production workloads.

To get started:

- [Get started with TmaxSoft OpenFrame](#)
- [Download the ebook](#)

## IBM zD&T 12.0

IBM Z Development and Test Environment (IBM zD&T) sets up a non-production environment on Azure that you can use for development, testing, and demos of z/OS-based applications.

The emulation environment on Azure can host different kinds of Z instances through Application Developers Controlled Distributions (ADCDs). You can run zD&T Personal Edition, zD&T Parallel Sysplex, and zD&T Enterprise Edition on Azure and Azure Stack.

To get started:

- [Set up IBM zD&T 12.0 on Azure](#)
- [Set up ADCD on zD&T](#)

## IBM DB2 pureScale on Azure

The IBM DB2 pureScale environment provides a database cluster for Azure. It's not identical to the original environment, but it delivers similar availability and scale as IBM DB2 for z/OS running in a Parallel Sysplex setup.

To get started, see [IBM DB2 pureScale on Azure](#).

## Considerations

When you migrate mainframe workloads to Azure infrastructure as a service (IaaS), you can choose from several types of on-demand, scalable computing resources, including Azure VMs. Azure offers a range of [Linux](#) and [Windows](#) VMs.

### Compute

Azure compute power compares favorably to a mainframe's capacity. If you're thinking of moving a mainframe workload to Azure, compare the mainframe metric of one million instructions per second (MIPS) to virtual CPUs.

Learn how to [move mainframe compute to Azure](#).

### High availability and failover

Azure offers commitment-based service-level agreements (SLAs). Multiple-nines availability is the default, and SLAs can be optimized with local or geo-based replication of services. The full [Azure SLA](#) explains the

guaranteed availability of Azure as a whole.

With Azure IaaS such as a VM, specific system functions provide failover support—for example, failover clustering instances and availability sets. When you use Azure platform as a service (PaaS) resources, the platform handles failover automatically. Examples include [Azure SQL Database](#) and [Azure Cosmos DB](#).

## Scalability

Mainframes typically scale up, while cloud environments scale out. Azure offers a range of [Linux](#) and [Windows](#) sizes to meet your needs. The cloud also scales up or down to match exact user specifications. Compute power, storage, and services [scale on demand](#) under a usage-based billing model.

## Storage

In the cloud, you have a range of flexible, scalable storage options, and you pay only for what you need. [Azure Storage](#) offers a massively scalable object store for data objects, a file system service for the cloud, a reliable messaging store, and a NoSQL store. For VMs, managed and unmanaged disks provide persistent, secure disk storage.

Learn how to [move mainframe storage to Azure](#).

## Backup and recovery

Maintaining your own disaster recovery site can be an expensive proposition. Azure has easy-to-implement and cost-effective options for [backup](#), [recovery](#), and [redundancy](#) at local or regional levels, or via geo-redundancy.

## Azure Government for mainframe migrations

Many public sector entities would love to move their mainframe applications to a more modern, flexible platform. Microsoft Azure Government is a physically separated instance of the global Microsoft Azure platform—packaged for federal, state, and local government systems. It provides world-class security, protection, and compliance services specifically for United States government agencies and their partners.

Azure Government earned a Provisional Authority to Operate (P-ATO) for FedRAMP High Impact for systems that need this type of environment.

To get started, download [Microsoft Azure Government cloud for mainframe applications](#).

## Next steps

Ask our [partners](#) to help you migrate or rehost your mainframe applications.

See also:

- [White papers about mainframe topics](#)
- [Mainframe migration](#)
- [Troubleshooting](#)
- [Demystifying mainframe to Azure migration](#)

# What is confidential computing?

9/21/2022 • 2 minutes to read • [Edit Online](#)

Confidential computing is an industry term defined by the [Confidential Computing Consortium \(CCC\)](#) - a foundation dedicated to defining and accelerating the adoption of confidential computing. The CCC defines confidential computing as: The protection of data in use by performing computations in a hardware-based Trusted Execution Environment (TEE).

A TEE is an environment that enforces execution of only authorized code. Any data in the TEE can't be read or tampered with by any code outside that environment. The confidential computing threat model aims at removing or reducing the ability for a cloud provider operator and other actors in the tenant's domain to access code and data while being executed.



When used with data encryption at rest and in transit, confidential computing eliminates the single largest barrier of encryption - encryption while in use - by protecting sensitive or highly regulated data sets and application workloads in a secure public cloud platform. Confidential computing extends beyond generic data protection. TEEs are also being used to protect proprietary business logic, analytics functions, machine learning algorithms, or entire applications.

## Lessen the need for trust

Running workloads on the cloud requires trust. You give this trust to various providers enabling different components of your application.

- **App software vendors:** Trust software by deploying on-premises, using open-source, or by building in-house application software.
- **Hardware vendors:** Trust hardware by using on-premises hardware or in-house hardware.
- **Infrastructure providers:** Trust cloud providers or manage your own on-premises data centers.

## Reducing the attack surface

The trusted computing base (TCB) refers to all of a system's hardware, firmware, and software components that provide a secure environment. The components inside the TCB are considered "critical". If one component inside the TCB is compromised, the entire system's security may be jeopardized. A lower TCB means higher security. There's less risk of exposure to various vulnerabilities, malware, attacks, and malicious people.

## Next steps

[Microsoft's offerings](#) for confidential computing extend from Infrastructure as a Service (IaaS) to Platform as a Service (PaaS) and as well as developer tools to support your journey to data and code confidentiality in the cloud. Learn more about confidential computing on Azure

[Overview of Azure Confidential Computing](#)

# Sizes for virtual machines in Azure

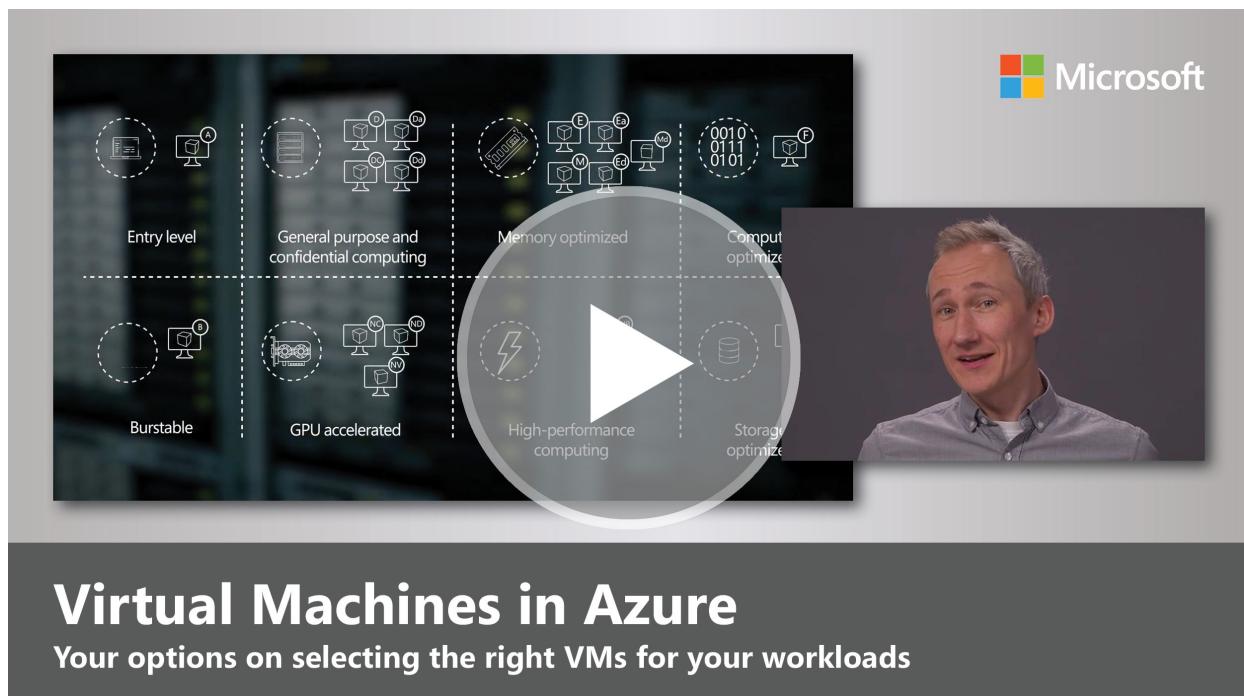
9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes the available sizes and options for the Azure virtual machines you can use to run your apps and workloads. It also provides deployment considerations to be aware of when you're planning to use these resources.

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.



## Virtual Machines in Azure

Your options on selecting the right VMs for your workloads

Type	Sizes	Description
General purpose	B, Dsv3, Dv3, Dasv4, Dav4, DSv2, Dv2, Av2, DC, DCv2, Dpdsv5, Dplds5, Dpsv5, Dplsv5, Dv4, Dsv4, Ddv4, Ddsv4, Dv5, Dsv5, Ddv5, Ddsv5, Dasv5, Dadsv5	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	F, Fs, Fsv2, FX	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	Esv3, Ev3, Easv4, Eav4, Epdsv5, Epsv5, Ev4, Esv4, Edv4, Edsv4, Ev5, Esv5, Edv5, Edsv5, Easv5, Eads5, Mv2, M, DSv2, Dv2	High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.

Type	Sizes	Description
Storage optimized	Lsv2, Lsv3, Lasv3	High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases.
GPU	NC, NCv2, NCv3, NCasT4_v3, ND, NDv2, NV, NVv3, NVv4, NDAsrA100_v4, NDm_A100_v4	Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs.
High performance compute	HB, HBv2, HBv3, HC, H	Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

- For information about pricing of the various sizes, see the pricing pages for [Linux](#) or [Windows](#).
- For availability of VM sizes in Azure regions, see [Products available by region](#).
- To see general limits on Azure VMs, see [Azure subscription and service limits, quotas, and constraints](#).
- For more information on how Azure names its VMs, see [Azure virtual machine sizes naming conventions](#).

## REST API

For information on using the REST API to query for VM sizes, see the following:

- [List available virtual machine sizes for resizing](#)
- [List available virtual machine sizes for a subscription](#)
- [List available virtual machine sizes in an availability set](#)

## ACU

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

## Benchmark scores

Learn more about compute performance for Linux VMs using the [CoreMark benchmark scores](#).

Learn more about compute performance for Windows VMs using the [SPECInt benchmark scores](#).

## Manage costs

Azure services cost money. Azure Cost Management helps you set budgets and configure alerts to keep spending under control. Analyze, manage, and optimize your Azure costs with Cost Management. To learn more, see the [quickstart on analyzing your costs](#).

## Next steps

Learn more about the different VM sizes that are available:

- [General purpose](#)
- [Compute optimized](#)
- [Memory optimized](#)

- Storage optimized
- GPU
- High performance compute
- Check the [Previous generation](#) page for A Standard, Dv1 (D1-4 and D11-14 v1), and A8-A11 series

# General purpose virtual machine sizes

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

General purpose VM sizes provide balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers. This article provides information about the offerings for general purpose computing.

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

- The [Av2-series](#) VMs can be deployed on various hardware types and processors. A-series VMs have CPU performance and memory configurations best suited for entry level workloads like development and test. This size is throttled, based on the hardware. The size offers consistent processor performance for the running instance, regardless of the hardware it's deployed on. To determine the physical hardware on which this size is deployed, query the virtual hardware from within the Virtual Machine. Example use cases include development and test servers, low traffic web servers, small to medium databases, proof-of-concepts, and code repositories.
- [B-series burstable](#) VMs are ideal for workloads that don't need the full performance of the CPU continuously, like web servers, small databases and development and test environments. These workloads typically have burstable performance requirements. The B-Series provides these customers the ability to purchase a VM size with a price conscious baseline performance that allows the VM instance to build up credits when the VM is utilizing less than its base performance. When the VM has accumulated credit, the VM can burst above the VM's baseline using up to 100% of the CPU when your application requires the higher CPU performance.
- The [DCv2-series](#) can help protect the confidentiality and integrity of your data and code while it's processed in the public cloud. These machines are backed by the latest generation of Intel XEON E-2288G Processor with SGX technology. With the Intel Turbo Boost Technology, these machines can go up to 5.0 GHz. DCv2 series instances enable customers to build secure enclave-based applications to protect their code and data while it's in use.
- The [Dpsv5 and Dpdsv5-series](#) and [Dplsv5 and Dpldsv5-series](#) are ARM64-based VMs featuring the 80 core, 3.0 GHz Ampere Altra processor. These series are designed for common enterprise workloads. They're optimized for database, in-memory caching, analytics, gaming, web, and application servers running on Linux.
- [Dv2 and Dsv2-series](#) VMs, a follow-on to the original D-series, features a more powerful CPU and optimal CPU-to-memory configuration making them suitable for most production workloads. The Dv2-series is about 35% faster than the D-series. Dv2-series run on 2nd Generation Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1 GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with the Intel Turbo Boost Technology 2.0. The Dv2-series has the same memory and disk configurations as the D-series.
- The [Dv3 and Dsv3-series](#) runs on 2nd Generation Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1 GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors. These series run in a hyper-threaded configuration, providing a better value proposition for most general purpose workloads. Memory has

been expanded (from ~3.5 GiB/vCPU to 4 GiB/vCPU) while disk and network limits have been adjusted on a per core basis to align with the move to hyperthreading. The Dv3-series no longer has the high memory VM sizes of the D/Dv2-series. Those sizes have been moved to the memory optimized [Ev3](#) and [Esv3-series](#).

- [Dav4 and Dasv4-series](#) are new sizes utilizing AMD's 2.35Ghz EPYC™ 7452 processor in a multi-threaded configuration with up to 256 MB L3 cache dedicating 8 MB of that L3 cache to every eight cores increasing customer options for running their general purpose workloads. The Dav4-series and Dasv4-series have the same memory and disk configurations as the D & Dsv3-series.
- The [Dv4 and Dsv4-series](#) runs on the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, providing a better value proposition for most general-purpose workloads. It features an all core Turbo clock speed of 3.4 GHz.
- The [Ddv4 and Ddsv4-series](#) runs on the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, providing a better value proposition for most general-purpose workloads. It features an all core Turbo clock speed of 3.4 GHz, [Intel® Turbo Boost Technology 2.0](#), [Intel® Hyper-Threading Technology](#) and [Intel® Advanced Vector Extensions 512 \(Intel® AVX-512\)](#). They also support [Intel® Deep Learning Boost](#). These new VM sizes will have 50% larger local storage, and better local disk IOPS for both read and write compared to the [Dv3/Dsv3](#) sizes with [Gen2 VMs](#).
- The [Dasv5 and Dadsv5-series](#) utilize AMD's 3rd Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache, increasing customer options for running their general purpose workloads. These virtual machines offer a combination of vCPUs and memory to meet the requirements associated with most enterprise workloads. For example, you can use these series with small-to-medium databases, low-to-medium traffic web servers, application servers, and more.
- The [Dv5 and Dsv5-series](#) run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a hyper-threaded configuration. The Dv5 and Dsv5 virtual machine sizes don't have any temporary storage thus lowering the price of entry. The Dv5 VM sizes offer a combination of vCPUs and memory to meet the requirements associated with most enterprise workloads. For example, you can use these series with small-to-medium databases, low-to-medium traffic web servers, application servers, and more.
- The [Ddv5 and Ddsv5-series](#) run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processors in a hyper-threaded configuration, providing a better value proposition for most general-purpose workloads. This new processor features an all core Turbo clock speed of 3.5 GHz, [Intel® Hyper-Threading Technology](#), [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#).

## Other sizes

- [Compute optimized](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

For more information on how Azure names its VMs, see [Azure virtual machine sizes naming conventions](#).

# Av1-series retirement

9/21/2022 • 2 minutes to read • [Edit Online](#)

On August 31, 2024, we retire Basic and Standard A-series virtual machines (VMs). Before that date, migrate your workloads to Av2-series VMs, which provide more memory per vCPU and faster storage on solid-state drives (SSDs).

## NOTE

In some cases, you must deallocate the VM prior to resizing. This can happen if the new size is not available on the hardware cluster that is currently hosting the VM.

## Migrate workloads to Av2-series VMs

You can resize your virtual machines to the Av2-series using the [Azure portal](#), [PowerShell](#), or the [CLI](#). Below are examples on how to resize your VM using the Azure portal and PowerShell.

## IMPORTANT

Resizing a virtual machine results in a restart. We recommend that you perform actions that result in a restart during off-peak business hours.

### Azure portal

1. Open the [Azure portal](#).
2. Type *virtual machines* in the search.
3. Under **Services**, select **Virtual machines**.
4. In the **Virtual machines** page, select the virtual machine you want to resize.
5. In the left menu, select **size**.
6. Pick a new Av2 size from the list of available sizes and select **Resize**.

### Azure PowerShell

1. Set the resource group and VM name variables. Replace the values with information of the VM you want to resize.

```
$resourceGroup = "myResourceGroup"  
$vmName = "myVM"
```

2. List the VM sizes that are available on the hardware cluster where the VM is hosted.

```
Get-AzVMSize -ResourceGroupName $resourceGroup -VMName $vmName
```

3. Resize the VM to the new size.

```
$vm = Get-AzVM -ResourceGroupName $resourceGroup -VMName $vmName  
$vm.HardwareProfile.VmSize = "<newAv2VmSize>"  
Update-AzVM -VM $vm -ResourceGroupName $resourceGroup
```

## Help and support

If you have questions, ask community experts in [Microsoft Q&A](#). If you have a support plan and need technical help, create a support request:

1. In the [Help + support](#) page, select **Create a support request**. Follow the [New support request](#) page instructions. Use the following values:
  - For **Issue type**, select **Technical**.
  - For **Service**, select **My services**.
  - For **Service type**, select **Virtual Machine running Windows/Linux**.
  - For **Resource**, select your VM.
  - For **Problem type**, select **Assistance with resizing my VM**.
  - For **Problem subtype**, select the option that applies to you.

Follow instructions in the **Solutions** and **Details** tabs, as applicable, and then **Review + create**.

## Next steps

Learn more about the [Av2-series VMs](#)

# Av2-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Av2-series VMs can be deployed on a variety of hardware types and processors. Av2-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake), the Intel® Xeon® 8171M 2.1 GHz (Skylake), the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors. Av2-series VMs have CPU performance and memory configurations best suited for entry level workloads like development and test. The size is throttled to offer consistent processor performance for the running instance, regardless of the hardware it is deployed on. To determine the physical hardware on which this size is deployed, query the virtual hardware from within the Virtual Machine. Some example use cases include development and test servers, low traffic web servers, small to medium databases, proof-of-concepts, and code repositories.

[ACU: 100](#)

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCORE	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THR OUGHPUT: IOPS	MAX NICs	EXPECTED NETWORK BANDWIDT H (MBPS)
Standard_A_1_v2	1	2	10	1000/20/10	2/2x500	2	250
Standard_A_2_v2	2	4	20	2000/40/20	4/4x500	2	500
Standard_A_4_v2	4	8	40	4000/80/40	8/8x500	4	1000
Standard_A_8_v2	8	16	80	8000/160/80	16/16x500	8	2000
Standard_A_2m_v2	2	16	20	2000/40/20	4/4x500	2	500
Standard_A_4m_v2	4	32	40	4000/80/40	8/8x500	4	1000

SIZE	VCORE	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THR OUGHPUT: IOPS	MAX NICs	EXPECTED NETWORK BANDWIDT H (MBPS)
Standard_A 8m_v2	8	64	80	8000/160/ 80	16/16x500	8	2000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# B-series burstable virtual machine sizes

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The B-series VMs can be deployed on a variety of hardware types and processors, so competitive bandwidth allocation is provided. B-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake), the Intel® Xeon® 8171M 2.1 GHz (Skylake), the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors. B-series VMs are ideal for workloads that do not need the full performance of the CPU continuously, like web servers, proof of concepts, small databases and development build environments. These workloads typically have burstable performance requirements. To determine the physical hardware on which this size is deployed, query the virtual hardware from within the virtual machine. The B-series provides you with the ability to purchase a VM size with baseline performance that can build up credits when it is using less than its baseline. When the VM has accumulated credits, the VM can burst above the baseline using up to 100% of the vCPU when your application requires higher CPU performance.

The B-series comes in the following VM sizes:

[Azure Compute Unit \(ACU\)](#): Varies\*

[Premium Storage](#): Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported\*\*

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

\*B-series VMs are burstable and thus ACU numbers will vary depending on workloads and core usage.

\*\*Accelerated Networking is only supported for *Standard\_B12ms*, *Standard\_B16ms* and *Standard\_B20ms*.

SIZE	VCP U	MEM ORY: GIB	TEM P STOR AGE (SSD) GIB	BASE CPU PERF OF VM	MAX CPU PERF OF VM	INITI AL CRED ITS	CRED ITS BAN KED/ HOU R	MAX BAN KED CRED ITS	MAX DATA DISK S	MAX BURS T UNC ACH ED DISK THR OUG HPU T: IOPS /MBP S	MAX UNC ACH ED DISK THR OUG HPU T: IOPS /MBP S <sup>1</sup>	MAX NICS
Standard_B1ls <sup>2</sup>	1	0.5	4	5%	100 %	30	3	72	2	160/10	4000/100	2
Standard_B1s	1	1	4	10%	100 %	30	6	144	2	320/10	4000/100	2

SIZE	VCP U	MEM ORY: GIB	TEM P STOR AGE (SSD) GIB	BASE CPU PERF OF VM	MAX CPU PERF OF VM	INITI AL CRED ITS	CRED ITS BAN KED/ HOU R	MAX BAN KED CRED ITS	MAX DATA DISK S	MAX BURS T UNC ACH ED DISK THR OUG HPU T: IOPS /MBP S	MAX UNC ACH ED DISK THR OUG HPU T: IOPS /MBP S	MAX NICS
------	----------	--------------------	---	---------------------------------	--------------------------------	----------------------------	--	----------------------------------	--------------------------	--	---	-------------

Standard_B1ms	1	2	4	20%	100 %	30	12	288	2	640/10	4000/100	2
Standard_B2s	2	4	8	40%	200 %	60	24	576	4	1280/15	4000/100	3
Standard_B2ms	2	8	16	60%	200 %	60	36	864	4	1920/22.5	4000/100	3
Standard_B4ms	4	16	32	90%	400 %	120	54	1296	8	2880/35	8000/200	4
Standard_B8ms	8	32	64	135 %	800 %	240	81	1944	16	4320/50	8000/200	4
Standard_B12ms	12	48	96	202 %	1200 %	360	121	2909	16	4320/50	1600/400	6
Standard_B16ms	16	64	128	270 %	1600 %	480	162	3888	32	4320/50	1600/400	8
Standard_B20ms	20	80	160	337 %	2000 %	600	203	4860	32	4320/50	1600/400	8

<sup>1</sup> B-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> B1Is is supported only on Linux

## Workload example

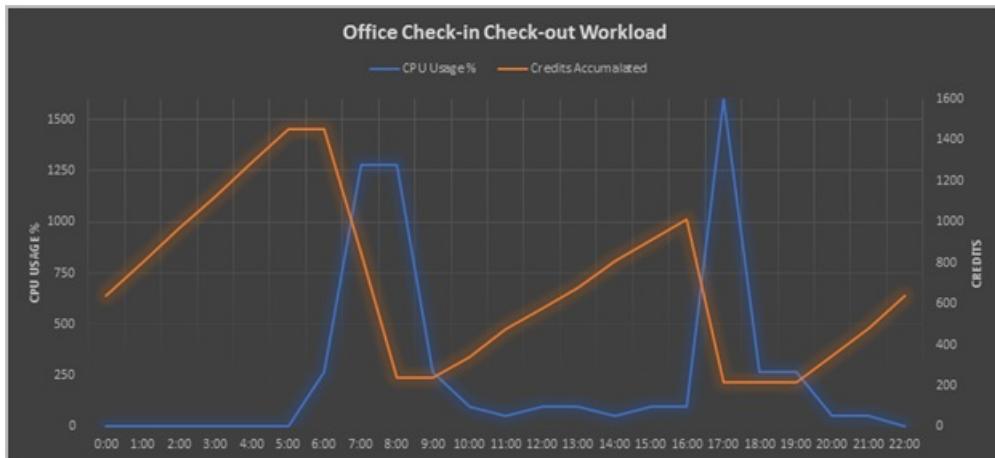
Consider an office check-in/out application. The application needs CPU bursts during business hours, but not a lot of computing power during off hours. In this example, the workload requires a 16vCPU virtual machine with 64GiB of RAM to work efficiently.

The table shows the hourly traffic data and the chart is a visual representation of that traffic.

B16 characteristics:

Max CPU perf:  $16\text{vCPU} * 100\% = 1600\%$

Baseline: 270%



SCENARIO	TIME	CPU USAGE (%)	CREDITS ACCUMULATED <sup>1</sup>	CREDITS AVAILABLE
B16ms Deployment	Deployment	Deployment	480 (Initial Credits)	480
No traffic	0:00	0	162	642
No traffic	1:00	0	162	804
No traffic	2:00	0	162	966
No traffic	3:00	0	162	1128
No traffic	4:00	0	162	1290
No traffic	5:00	0	162	1452
Low Traffic	6:00	270	0	1452
Employees come to office (app needs 80% vCPU)	7:00	1280	-606	846
Employees continue coming to office (app needs 80% vCPU)	8:00	1280	-606	240
Low Traffic	9:00	270	0	240

SCENARIO	TIME	CPU USAGE (%)	CREDITS ACCUMULATED	CREDITS AVAILABLE
Low Traffic	10:00	100	102	342
Low Traffic	11:00	50	132	474
Low Traffic	12:00	100	102	576
Low Traffic	13:00	100	102	678
Low Traffic	14:00	50	132	810
Low Traffic	15:00	100	102	912
Low Traffic	16:00	100	102	1014
Employees checking out (app needs 100% vCPU)	17:00	1600	-798	216
Low Traffic	18:00	270	0	216
Low Traffic	19:00	270	0	216
Low Traffic	20:00	50	132	348
Low Traffic	21:00	50	132	480
No traffic	22:00	0	162	642
No traffic	23:00	0	162	804

<sup>1</sup> Credits accumulated/credits used in an hour is equivalent to:

$$((\text{Base CPU perf of VM} - \text{CPU Usage}) / 100) * 60 \text{ minutes}$$

For a D16s\_v3 which has 16 vCPUs and 64 GiB of memory the hourly rate is \$0.936 per hour (monthly \$673.92) and for B16ms with 16 vCPUs and 64 GiB memory the rate is \$0.794 per hour (monthly \$547.86). This results in 15% savings!

## Q & A

### Q: What happens when my credits run out?

A: When the credits are exhausted, the VM returns to the baseline performance.

### Q: How do you get 135% baseline performance from a VM?

A: The 135% is shared amongst the 8 vCPU's that make up the VM size. For example, if your application uses 4 of the 8 cores working on batch processing and each of those 4 vCPU's are running at 30% utilization the total amount of VM CPU performance would equal 120%. Meaning that your VM would be building credit time based on the 15% delta from your baseline performance. But it also means that when you have credits available that same VM can use 100% of all 8 vCPU's giving that VM a Max CPU performance of 800%.

### Q: How can I monitor my credit balance and consumption?

A: The Credit metric allows you to view how many credits your VM have been banked and the

**ConsumedCredit** metric will show how many CPU credits your VM has consumed from the bank. You will be able to view these metrics from the metrics pane in the portal or programmatically through the Azure Monitor APIs.

For more information on how to access the metrics data for Azure, see [Overview of metrics in Microsoft Azure](#).

**Q: How are credits accumulated and consumed?**

A: The VM accumulation and consumption rates are set such that a VM running at exactly its base performance level will have neither a net accumulation or consumption of bursting credits. A VM will have a net increase in credits whenever it is running below its base performance level and will have a net decrease in credits whenever the VM is utilizing the CPU more than its base performance level.

**Example:** I deploy a VM using the B1ms size for my small time and attendance database application. This size allows my application to use up to 20% of a vCPU as my baseline, which is 0.2 credits per minute I can use or bank.

My application is busy at the beginning and end of my employees work day, between 7:00-9:00 AM and 4:00 - 6:00PM. During the other 20 hours of the day, my application is typically at idle, only using 10% of the vCPU. For the non-peak hours, I earn 0.2 credits per minute but only consume 0.1 credits per minute, so my VM will bank  $0.1 \times 60 = 6$  credits per hour. For the 20 hours that I am off-peak, I will bank 120 credits.

During peak hours my application averages 60% vCPU utilization, I still earn 0.2 credits per minute but I consume 0.6 credits per minute, for a net cost of 0.4 credits a minute or  $0.4 \times 60 = 24$  credits per hour. I have 4 hours per day of peak usage, so it costs  $4 \times 24 = 96$  credits for my peak usage.

If I take the 120 credits I earned off-peak and subtract the 96 credits I used for my peak times, I bank an additional 24 credits per day that I can use for other bursts of activity.

**Q: How can I calculate credits accumulated and used?**

A: You can use the following formula:

$(\text{Base CPU perf of VM} - \text{CPU Usage}) / 100 = \text{Credits bank or use per minute}$

e.g in above instance your baseline is 20% and if you use 10% of the CPU you are accumulating  $(20\%-10\%)/100 = 0.1$  credit per minute.

**Q: Does the B-Series support Premium Storage data disks?**

A: Yes, all B-Series sizes support Premium Storage data disks.

**Q: Why is my remaining credit set to 0 after a redeploy or a stop/start?**

A : When a VM is redeployed and the VM moves to another node, the accumulated credit is lost. If the VM is stopped/started, but remains on the same node, the VM retains the accumulated credit. Whenever the VM starts fresh on a node, it gets an initial credit, for Standard\_B8ms it is 240.

**Q: What happens if I deploy an unsupported OS image on B1ls?**

A : B1ls only supports Linux images and if you deploy any another OS image you might not get the best customer experience.

## Other sizes and information

- [General purpose](#)
- [Compute optimized](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# DCsv2-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The DCsv2-series virtual machines help protect the confidentiality and integrity of your data and code while it's processed in the public cloud. DCsv2-series leverage Intel® Software Guard Extensions, which enable customers to use secure enclaves for protection.

These machines are backed by 3.7 GHz Intel® Xeon E-2288G (Coffee Lake) with SGX technology. With Intel® Turbo Boost Max Technology 3.0 these machines can go up to 5.0 GHz.

## NOTE

Hyperthreading is disabled for added security posture. Pricing is the same as Dv5 and Dsv5-series per physical core.

Example confidential use cases include: databases, blockchain, multiparty data analytics, fraud detection, anti-money laundering, usage analytics, intelligence analysis and machine learning.

## Configuration

[Turbo Boost Max 3.0](#): Supported (Tenant VM will report 3.7 GHz, but will reach Turbo Speeds)

[Hyper-Threading](#): Not Supported

[Premium Storage](#): Supported (Not Supported for Standard\_DC8\_v2)

[Premium Storage Caching](#): Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 2

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

## Technical specifications

SIZE	PHYSICAL CORES	MEMORY GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX NICs	EPC MEMORY MIB
Standard_DC 1s_v2	1	4	50	1	1	28
Standard_DC 2s_v2	2	8	100	2	1	56
Standard_DC 4s_v2	4	16	200	4	1	112
Standard_DC 8_v2	8	32	400	8	1	168

## Get started

- Create DCsv2 VMs using the [Azure portal](#) or [Azure Marketplace](#)
- DCsv2-series VMs are [Generation 2 VMs](#) and only support [Gen2](#) images.
- Currently available in the regions listed in [Azure Products by Region](#).

## More sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)
- [Pricing Calculator](#)
- [More On Disk Types](#)

Pricing Calculator : [Pricing Calculator](#)

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# DCsv3 and DCcsv3-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The DCsv3 and DCcsv3-series Azure Virtual Machines help protect the confidentiality and integrity of your code and data while they're being processed in the public cloud. By using Intel® Software Guard Extensions and Intel® Total Memory Encryption - Multi Key, customers can ensure their data is always encrypted and protected in use.

These machines are powered by the latest 3rd Generation Intel® Xeon Scalable processors, and use Intel® Turbo Boost Max Technology 3.0 to reach 3.5 GHz.

With this generation, CPU Cores have increased 6x (up to a maximum of 48 physical cores). Encrypted Memory (EPC) has increased 1500x to 256 GB. Regular Memory has increased 12x to 384 GB. All these changes substantially improve the performance and unlock new entirely new scenarios.

## NOTE

Hyperthreading is disabled for added security posture. Pricing is the same as Dv5 and Dsv5-series per physical core.

There are two variants for each series, depending on whether the workload benefits from a local disk or not. You can attach remote persistent disk storage to all VMs, whether or not the VM has a local disk. As always, remote disk options (such as for the VM boot disk) are billed separately from the VMs in any case.

DCsv3-series instances run on a 3rd Generation Intel® Xeon Scalable Processor 8370C. The base All-Core frequency is 2.8 GHz. [Turbo Boost Max 3.0](#) is enabled with a max frequency of 3.5 GHz.

- [Premium Storage](#): Supported
- [Live Migration](#): Not supported
- [Memory Preserving Updates](#): Not supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported for DCcsv3-series
- [Ultra-Disk Storage](#): Supported
- [Azure Kubernetes Service](#): Supported (CLI provisioning only)
- [Nested Virtualization](#): Not Supported
- [Hyper-Threading](#): Not supported
- [Trusted Launch](#): Supported
- [Dedicated Host](#): Not supported

## DCsv3-series

SIZE	PHYSICAL CORES	MEMORY GB	TEMP STORAGE (SSD) GB	MAX DATA DISKS	MAX NICs	EPC MEMORY GB
Standard_DC 1s_v3	1	8	Remote Storage Only	4	2	4

SIZE	PHYSICAL CORES	MEMORY GB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX NICs	EPC MEMORY GIB
Standard_DC 2s_v3	2	16	Remote Storage Only	8	2	8
Standard_DC 4s_v3	4	32	Remote Storage Only	16	4	16
Standard_DC 8s_v3	8	64	Remote Storage Only	32	8	32
Standard_DC 16s_v3	16	128	Remote Storage Only	32	8	64
Standard_DC 24s_v3	24	192	Remote Storage Only	32	8	128
Standard_DC 32s_v3	32	256	Remote Storage Only	32	8	192
Standard_DC 48s_v3	48	384	Remote Storage Only	32	8	256

## DCcsv3-series

SIZE	PHYSICAL CORES	MEMORY GB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX NICs	EPC MEMORY GIB
Standard_DC 1ds_v3	1	8	75	4	2	4
Standard_DC 2ds_v3	2	16	150	8	2	8
Standard_DC 4ds_v3	4	32	300	16	4	16
Standard_DC 8ds_v3	8	64	600	32	8	32
Standard_DC 16ds_v3	16	128	1200	32	8	64
Standard_DC 24ds_v3	24	192	1800	32	8	128
Standard_DC 32ds_v3	32	256	2400	32	8	192
Standard_DC 48ds_v3	48	384	2400	32	8	256

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## More sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)
- [Pricing Calculator](#)

## Next steps

- Create DCsv3 and DCdsv3 VMs using the [Azure portal](#)
- DCsv3 and DCdsv3 VMs are [Generation 2 VMs](#) and only support [Gen2](#) images.
- Currently available in the regions listed in [Azure Products by Region](#).

Pricing Calculator: [Pricing Calculator](#)

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dv2 and DSv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dv2 and DSv2-series, a follow-on to the original D-series, feature a more powerful CPU and optimal CPU-to-memory configuration making them suitable for most production workloads. The Dv2-series is about 35% faster than the D-series. Dv2-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with the Intel Turbo Boost Technology 2.0. The Dv2-series has the same memory and disk configurations as the D-series.

## Dv2-series

Dv2-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake), or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with Intel Turbo Boost Technology 2.0.

[ACU](#): 210-250

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS	THROUGHPUT: IOPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D1_v2 <sup>1</sup>	1	3.5	50	3000/46/23	4	4x500	2	750
Standard_D2_v2	2	7	100	6000/93/46	8	8x500	2	1500
Standard_D3_v2	4	14	200	12000/187/93	16	16x500	4	3000
Standard_D4_v2	8	28	400	24000/375/187	32	32x500	8	6000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUG HPUT: IOPS/REA D MBPS/WR ITE MBPS	MAX DATA DISKS	THROUG HPUT: IOPS	MAX NICS	EXPECTE D NETWOR K BANDWID TH (MBPS)
Standard_ D5_v2	16	56	800	48000/75 0/375	64	64x500	8	12000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## DSv2-series

DSv2-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake) or the the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with Intel Turbo Boost Technology 2.0 and use premium storage.

[ACU](#): 210-250

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUG HPUT: IOPS/MB PS (CACHE SIZE IN GIB)	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	EXPECTE D NETWOR K BANDWID TH (MBPS)
Standard_ DS1_v2 <sup>1</sup>	1	3.5	7	4	4000/32 (43)	3200/48	2	750
Standard_ DS2_v2	2	7	14	8	8000/64 (86)	6400/96	2	1500
Standard_ DS3_v2	4	14	28	16	16000/12 8 (172)	12800/19 2	4	3000
Standard_ DS4_v2	8	28	56	32	32000/25 6 (344)	25600/38 4	8	6000
Standard_ DS5_v2	16	56	112	64	64000/51 2 (688)	51200/76 8	8	12000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dv3 and Dsv3-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dv3-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors in a hyper-threaded configuration, providing a better value proposition for most general purpose workloads. Memory has been expanded (from ~3.5 GiB/vCPU to 4 GiB/vCPU) while disk and network limits have been adjusted on a per core basis to align with the move to hyperthreading. The Dv3-series no longer has the high memory VM sizes of the D/Dv2-series, those have been moved to the memory optimized [Ev3 and Esv3-series](#).

Example D-series use cases include enterprise-grade applications, relational databases, in-memory caching, and analytics.

## Dv3-series

Dv3-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with [Intel® Turbo Boost Technology 2.0](#). The Dv3-series sizes offer a combination of vCPU, memory, and temporary storage for most production workloads.

Data disk storage is billed separately from virtual machines. To use premium storage disks, use the Dsv3 sizes. The pricing and billing meters for Dsv3 sizes are the same as Dv3-series.

Dv3-series VMs feature Intel® Hyper-Threading Technology.

[ACU](#): 160-190

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX NICs/EXPECTED NETWORK BANDWIDTH
Standard_D2_v3 <sup>1</sup>	2	8	50	4	3000/46/23	2/1000
Standard_D4_v3	4	16	100	8	6000/93/46	2/2000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX NICs/EXPECTED NETWORK BANDWIDTH
Standard_D8_v3	8	32	200	16	12000/187/93	4/4000
Standard_D16_v3	16	64	400	32	24000/375/187	8/8000
Standard_D32_v3	32	128	800	32	48000/750/375	8/16000
Standard_D48_v3	48	192	1200	32	96000/1000/500	8/24000
Standard_D64_v3	64	256	1600	32	96000/1000/500	8/30000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Dsv3-series

Dsv3-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1GHz (Skylake), Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors with [Intel® Turbo Boost Technology 2.0](#) and use premium storage. The Dsv3-series sizes offer a combination of vCPU, memory, and temporary storage for most production workloads.

Dsv3-series VMs feature Intel® Hyper-Threading Technology.

[ACU](#): 160-190

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS (CACHE SIZE IN GiB)	MAX BURST CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS <sup>2</sup>	MAX UNCACHED DISK THROUGHPUT: IOPS/M BPS	MAX BURST UNCAC HED DISK THROUGHPUT: IOPS/M BPS <sup>1</sup>	MAX NICS/EXPECT ED NETWORK BANDWIDTH (MBPS)
Standard_D2s_v3 <sup>2</sup>	2	8	16	4	4000/32 (50)	4000/200	3200/48	4000/200	2/1000
Standard_D4s_v3	4	16	32	8	8000/64 (100)	8000/200	6400/96	8000/200	2/2000
Standard_D8s_v3	8	32	64	16	16000/128 (200)	16000/400	12800/192	16000/400	4/4000
Standard_D16s_v3	16	64	128	32	32000/256 (400)	32000/800	25600/384	32000/800	8/8000
Standard_D32s_v3	32	128	256	32	64000/512 (800)	64000/1600	51200/768	64000/1600	8/1600
Standard_D48s_v3	48	192	384	32	96000/768 (1200)	96000/2000	76800/1152	80000/2000	8/2400
Standard_D64s_v3	64	256	512	32	128000/1024 (1600)	128000/2000	80000/1200	80000/2000	8/3000

<sup>1</sup> Dsv3-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).

- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dv4 and Dsv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dv4 and Dsv4-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, providing a better value proposition for most general-purpose workloads. It features an all core Turbo clock speed of 3.4 GHz, Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology and Intel® Advanced Vector Extensions 512 (Intel® AVX-512). They also support Intel® Deep Learning Boost.

## NOTE

For frequently asked questions, see [Azure VM sizes with no local temp disk](#).

## Dv4-series

Dv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake). The Dv4-series sizes offer a combination of vCPU, memory and remote storage options for most production workloads. Dv4-series VMs feature Intel® Hyper-Threading Technology.

Remote Data disk storage is billed separately from virtual machines. To use premium storage disks, use the Dsv4 sizes. The pricing and billing meters for Dsv4 sizes are the same as Dv4-series.

## NOTE

After a restart, a file named *Data\_loss\_warning.txt* might appear beside drive C (the first data disk attached from the Azure portal). In this scenario, despite the file name, no data loss has occurred on the disk. In general, the *Data\_loss\_warning.txt* file usually is copied on the temporary drive. If you're using a VM that doesn't have a temp drive, WindowsAzureGuestAgent incorrectly copies the file to the first drive letter. In v4 VMs, the first drive letter is a data disk.

A resolution for this issue was applied in the latest version (version 2.7.41491.999) of the VM agent.

**ACU:** 195-210

**Premium Storage:** Not Supported

**Premium Storage caching:** Not Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Not Supported

**Nested Virtualization:** Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_D2_v4 <sup>1</sup>	2	8	Remote Storage Only	4	2	5000

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D4_v4	4	16	Remote Storage Only	8	2	10000
Standard_D8_v4	8	32	Remote Storage Only	16	4	12500
Standard_D16_v4	16	64	Remote Storage Only	32	8	12500
Standard_D32_v4	32	128	Remote Storage Only	32	8	16000
Standard_D48_v4	48	192	Remote Storage Only	32	8	24000
Standard_D64_v4	64	256	Remote Storage Only	32	8	30000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Dsv4-series

Dsv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake). The Dv4-series sizes offer a combination of vCPU, memory and remote storage options for most production workloads. Dsv4-series VMs feature [Intel® Hyper-Threading Technology](#). Remote Data disk storage is billed separately from virtual machines.

[ACU](#): 195-210

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS <sup>1</sup>	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D2s_v4 <sup>2</sup>	2	8	Remote Storage Only	4	3200/48	4000/200	2	5000

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	EXPECTE D NETWOR K BANDWID TH (MBPS)
Standard_D4s_v4	4	16	Remote Storage Only	8	6400/96	8000/200	2	10000
Standard_D8s_v4	8	32	Remote Storage Only	16	12800/192	16000/400	4	12500
Standard_D16s_v4	16	64	Remote Storage Only	32	25600/384	32000/800	8	12500
Standard_D32s_v4	32	128	Remote Storage Only	32	51200/768	64000/1600	8	16000
Standard_D48s_v4	48	192	Remote Storage Only	32	76800/1152	80000/2000	8	24000
Standard_D64s_v4	64	256	Remote Storage Only	32	80000/1200	80000/2000	8	30000

<sup>1</sup> Dsv4-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize](#)

network throughput for Azure virtual machines. To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dav4 and Dasv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dav4-series and Dasv4-series run on 2nd Generation AMD EPYC™ 7452 or 3rd Generation EPYC™ 7763v processors in a multi-threaded configuration. The Dav4-series and Dasv4-series have the same memory and disk configurations as the D & Dsv3-series.

## Dav4-series

The Dav4-series run on 2nd Generation AMD EPYC™ 7452 (up to 3.35GHz) or 3rd Generation EPYC™ 7763v processors (up to 3.5GHz). The Dav4-series sizes offer a combination of vCPU, memory and temporary storage for most production workloads. Data disk storage is billed separately from virtual machines. To use premium SSD, use the Dasv4 sizes. The pricing and billing meters for Dasv4 sizes are the same as the Dav4-series.

[ACU: 230-260](#)

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D 2a_v4 <sup>1</sup>	2	8	50	4	3000 / 46 / 23	2	2000
Standard_D 4a_v4	4	16	100	8	6000 / 93 / 46	2	4000
Standard_D 8a_v4	8	32	200	16	12000 / 187 / 93	4	8000
Standard_D 16a_v4	16	64	400	32	24000 / 375 / 187	8	10000
Standard_D 32a_v4	32	128	800	32	48000 / 750 / 375	8	16000
Standard_D 48a_v4	48	192	1200	32	96000 / 1000 / 500	8	24000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D64a_v4	64	256	1600	32	96000 / 1000 / 500	8	32000
Standard_D96a_v4	96	384	2400	32	96000 / 1000 / 500	8	40000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Dasv4-series

The Dasv4-series run on 2nd Generation AMD EPYC™ 7452 (up to 3.35GHz) or 3rd Generation EPYC™ 7763v processors (up to 3.5GHz) and use premium SSD. The Dasv4-series sizes offer a combination of vCPU, memory and temporary storage for most production workloads.

[ACU](#): 230-260

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORAGETHROUGHPUT: IOPS / MBPS (CACHE SIZE IN GIB)	MAX BURST CACHE D AND TEMP STORAGETHROUGHPUT: IOPS / MBPS <sup>1</sup>	MAX UNCA CHED DISK THROUGHPUT: IOPS / MBPS	MAX BURST UNCA CHED DISK THROUGHPUT: IOPS/ MBPS <sup>1</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D2a_s_v4 <sup>2</sup>	2	8	16	4	4000 / 32 (50)	4000/100	3200 / 48	4000/200	2	2000
Standard_D4a_s_v4	4	16	32	8	8000 / 64 (100)	8000/200	6400 / 96	8000/200	2	4000
Standard_D8a_s_v4	8	32	64	16	16000 / 128 (200)	16000/400	12800 / 192	16000/400	4	8000

SIZE	VCPU	MEMO RY: GIB	TEMP STORA GE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORA GE THROU GHPUT : IOPS / MBPS (CACH E SIZE IN GIB)	MAX BURST CACHE D AND TEMP STORA GE THROU GHPUT : IOPS / MBPS	MAX UNCA CHED DISK THROU GHPUT : IOPS / MBPS	MAX BURST UNCA CHED DISK THROU GHPUT : IOPS/ MBPS	MAX NICS	EXPEC TED NETW ORK BAND WIDTH (MBPS)
Stand ard_D1 6as_v4	16	64	128	32	32000 / 255 (400)	32000/ 800	25600 / 384	32000/ 800	8	10000
Stand ard_D3 2as_v4	32	128	256	32	64000 / 510 (800)	64000/ 1600	51200 / 768	64000/ 1600	8	16000
Stand ard_D4 8as_v4	48	192	384	32	96000 / 1020 (1200)	96000/ 2000	76800 / 1148	80000/ 2000	8	24000
Stand ard_D6 4as_v4	64	256	512	32	12800 0 / 1020 (1600)	12800 0/2000	80000 / 1200	80000/ 2000	8	32000
Stand ard_D9 6as_v4	96	384	768	32	19200 0 / 1020 (2400)	19200 0/2000	80000 / 1200	80000/ 2000	8	40000

<sup>1</sup> Dasv4-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion,

application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Ddv4 and Ddsv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Ddv4 and Ddsv4-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, providing a better value proposition for most general-purpose workloads. It features an all core Turbo clock speed of 3.4 GHz, Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology and Intel® Advanced Vector Extensions 512 (Intel® AVX-512). They also support Intel® Deep Learning Boost. These new VM sizes will have 50% larger local storage, as well as better local disk IOPS for both read and write compared to the Dv3/Dsv3 sizes with Gen2 VMs.

D-series use cases include enterprise-grade applications, relational databases, in-memory caching, and analytics.

## Ddv4-series

Ddv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake). The Ddv4-series offer a combination of vCPU, memory and temporary disk for most production workloads.

The new Ddv4 VM sizes include fast, larger local SSD storage (up to 2,400 GiB) and are designed for applications that benefit from low latency, high-speed local storage, such as applications that require fast reads/ writes to temp storage or that need temp storage for caches or temporary files. You can attach Standard SSDs and Standard HDDs storage to the Ddv4 VMs. Remote Data disk storage is billed separately from virtual machines.

[ACU: 195-210](#)

[Premium Storage: Not Supported](#)

[Premium Storage caching: Not Supported](#)

[Live Migration: Supported](#)

[Memory Preserving Updates: Supported](#)

[VM Generation Support: Generation 1 and 2](#)

[Accelerated Networking: Supported](#)

[Ephemeral OS Disks: Supported](#)

[Nested Virtualization: Supported](#)

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/Mbps *	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_D2d_v4 <sup>1</sup>	2	8	75	4	9000/125	2	5000
Standard_D4d_v4	4	16	150	8	19000/250	2	10000
Standard_D8d_v4	8	32	300	16	38000/500	4	12500

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_D 16d_v4	16	64	600	32	75000/100 0	8	12500
Standard_D 32d_v4	32	128	1200	32	150000/20 00	8	16000
Standard_D 48d_v4	48	192	1800	32	225000/30 00	8	24000
Standard_D 64d_v4	64	256	2400	32	300000/40 00	8	30000

\* These IOPs values can be achieved by using [Gen2 VMs](#)

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Ddsv4-series

Ddsv4-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake). The Ddsv4-series offer a combination of vCPU, memory and temporary disk for most production workloads.

The new Ddsv4 VM sizes include fast, larger local SSD storage (up to 2,400 GiB) and are designed for applications that benefit from low latency, high-speed local storage, such as applications that require fast reads/writes to temp storage or that need temp storage for caches or temporary files.

### NOTE

The pricing and billing meters for Ddsv4 sizes are the same as Ddv4-series.

**ACU:** 195-210

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS*	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS <sup>1</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D2ds_v4 <sup>2</sup>	2	8	75	4	9000/125	3200/48	4000/200	2	5000
Standard_D4ds_v4	4	16	150	8	19000/250	6400/96	8000/200	2	10000
Standard_D8ds_v4	8	32	300	16	38000/500	12800/192	16000/400	4	12500
Standard_D16ds_v4	16	64	600	32	85000/1000	25600/384	32000/800	8	12500
Standard_D32ds_v4	32	128	1200	32	150000/2000	51200/768	64000/1600	8	16000
Standard_D48ds_v4	48	192	1800	32	225000/3000	76800/1152	80000/2000	8	24000
Standard_D64ds_v4	64	256	2400	32	300000/4000	80000/1200	80000/2000	8	30000

\* These IOPs values can be achieved by using [Gen2 VMs](#)

<sup>1</sup> Ddsv4-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all

NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dv5 and Dsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dv5 and Dsv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper threaded](#) configuration, providing a better value proposition for most general-purpose workloads. This new processor features an all core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#). These virtual machines offer a combination of vCPUs and memory to meet the requirements associated with most enterprise workloads, such as small-to-medium databases, low-to-medium traffic web servers, application servers and more. The Dv5 and Dsv5-series provide a better value proposition for workloads that don't require local temp disk. For information about similar virtual machines with local disk, see [Ddv5 and Ddsv5-series VMs](#).

## NOTE

For frequently asked questions, see [Azure VM sizes with no local temp disk](#).

## Dv5-series

Dv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 96 vCPU and 384 GiB of RAM. Dv5-series virtual machines provide a better value proposition for most general-purpose workloads compared to the prior generation (for example, increased scalability and an upgraded CPU class).

Dv5-series virtual machines do not have any temporary storage thus lowering the price of entry. You can attach Standard SSDs, and Standard HDDs disk storage to these virtual machines. To use Premium SSD or Ultra Disk storage, select Dsv5-series virtual machines. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	MAX NETWORK BANDWIDTH (Mbps)
Standard_D2_v5 <sup>1, 2</sup>	2	8	Remote Storage Only	4	2	12500
Standard_D4_v5	4	16	Remote Storage Only	8	2	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	MAX NETWORK BANDWIDTH (Mbps)
Standard_D8_v5	8	32	Remote Storage Only	16	4	12500
Standard_D16_v5	16	64	Remote Storage Only	32	8	12500
Standard_D32_v5	32	128	Remote Storage Only	32	8	16000
Standard_D48_v5	48	192	Remote Storage Only	32	8	24000
Standard_D64_v5	64	256	Remote Storage Only	32	8	30000
Standard_D96_v5	96	384	Remote Storage Only	32	8	35000

<sup>1</sup> Accelerated networking is required and turned on by default on all Dv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

## Dsv5-series

Dsv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 96 vCPU and 384 GiB of RAM. Dsv5-series virtual machines provide a better value proposition for most general-purpose workloads compared to the prior generation (for example, increased scalability and an upgraded CPU class).

Dsv5-series virtual machines do not have any temporary storage thus lowering the price of entry. You can attach Standard SSDs, Standard HDDs, and Premium SSDs disk storage to these virtual machines. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Required

**Ephemeral OS Disks:** Not Supported

**Nested Virtualization:** Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS <sup>3</sup>	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_D2s_v5 <sup>1,2</sup>	2	8	Remote Storage Only	4	3750/85	10000/1200	2	12500
Standard_D4s_v5	4	16	Remote Storage Only	8	6400/145	20000/1200	2	12500
Standard_D8s_v5	8	32	Remote Storage Only	16	12800/290	20000/1200	4	12500
Standard_D16s_v5	16	64	Remote Storage Only	32	25600/600	40000/1200	8	12500
Standard_D32s_v5	32	128	Remote Storage Only	32	51200/865	80000/2000	8	16000
Standard_D48s_v5	48	192	Remote Storage Only	32	76800/1315	80000/3000	8	24000
Standard_D64s_v5	64	256	Remote Storage Only	32	80000/1735	80000/3000	8	30000
Standard_D96s_v5	96	384	Remote Storage Only	32	80000/2600	80000/4000	8	35000

<sup>1</sup> Accelerated networking is required and turned on by default on all Dsv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> Dsv5-series virtual machines can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

# Ddv5 and Ddsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Ddv5 and Ddsv5-series Virtual Machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper threaded](#) configuration, providing a better value proposition for most general-purpose workloads. This new processor features an all core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#). These virtual machines offer a combination of vCPUs, memory and temporary storage able to meet the requirements associated with most enterprise workloads, such as small-to-medium databases, low-to-medium traffic web servers, application servers and more.

## Ddv5-series

Ddv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 96 vCPU and 384 GiB of RAM as well as fast, local SSD storage up to 3,600 GiB. Ddv5-series virtual machines provide a better value proposition for most general-purpose workloads compared to the prior generation (for example, increased scalability and an upgraded CPU class). These virtual machines also feature fast and large local SSD storage (up to 3,600 GiB).

Ddv5-series virtual machines support Standard SSD and Standard HDD disk types. To use Premium SSD or Ultra Disk storage, select Ddsv5-series virtual machines. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/Mbps *	MAX NICS	MAX NETWORK BANDWIDTH (Mbps)
Standard_D 2d_v5 <sup>1,2</sup>	2	8	75	4	9000/125	2	12500
Standard_D 4d_v5	4	16	150	8	19000/250	2	12500
Standard_D 8d_v5	8	32	300	16	38000/500	4	12500

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX NICS	MAX NETWORK BANDWIDT H (MBPS)
Standard_D 16d_v5	16	64	600	32	75000/100 0	8	12500
Standard_D 32d_v5	32	128	1200	32	150000/20 00	8	16000
Standard_D 48d_v5	48	192	1800	32	225000/30 00	8	24000
Standard_D 64d_v5	64	256	2400	32	300000/40 00	8	30000
Standard_D 96d_v5	96	384	3600	32	450000/40 00	8	35000

\* These IOPs values can be guaranteed by using [Gen2 VMs](#)

<sup>1</sup> Accelerated networking is required and turned on by default on all Ddv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

## Ddsv5-series

Ddsv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 96 vCPU and 384 GiB of RAM as well as fast, local SSD storage up to 3,600 GiB. Ddsv5-series virtual machines provide a better value proposition for most general-purpose workloads compared to the prior generation (for example, increased scalability and an upgraded CPU class).

Ddsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS*	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS <sup>3</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_D2ds_v5 <sup>1,2</sup>	2	8	75	4	9000/125	3750/85	10000/1200	2	12500
Standard_D4ds_v5	4	16	150	8	19000/250	6400/145	20000/1200	2	12500
Standard_D8ds_v5	8	32	300	16	38000/500	12800/290	20000/1200	4	12500
Standard_D16ds_v5	16	64	600	32	75000/1000	25600/600	40000/1200	8	12500
Standard_D32ds_v5	32	128	1200	32	150000/2000	51200/865	80000/2000	8	16000
Standard_D48ds_v5	48	192	1800	32	225000/3000	76800/1315	80000/3000	8	24000
Standard_D64ds_v5	64	256	2400	32	375000/4000	80000/1735	80000/3000	8	30000
Standard_D96ds_v5	96	384	3600	32	450000/4000	80000/2600	80000/4000	8	35000

\* These IOPs values can be guaranteed by using [Gen2 VMs](#)

<sup>1</sup> Accelerated networking is required and turned on by default on all Ddsv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> Ddsv5-series virtual machines can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to

## None.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

# Dasv5 and Dadsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dasv5-series and Dadsv5-series utilize AMD's 3rd Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache, increasing customer options for running their general purpose workloads. These virtual machines offer a combination of vCPUs and memory to meet the requirements associated with most enterprise workloads, such as small-to-medium databases, low-to-medium traffic web servers, application servers and more.

## Dasv5-series

Dasv5-series VMs utilize AMD's 3rd Generation EPYC™ 7763v processors that can achieve a boosted maximum frequency of 3.5GHz. The Dasv5-series sizes offer a combination of vCPU and memory for most production workloads. The new VMs with no local disk provide a better value proposition for workloads that do not require local temp disk.

### NOTE

For frequently asked questions, see [Azure VM sizes with no local temp disk](#).

Dasv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Not Supported

**Nested Virtualization:** Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS <sup>1</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_D2as_v5	2	8	Remote Storage Only	4	3750/82	10000/600	2	12500
Standard_D4as_v5	4	16	Remote Storage Only	8	6400/144	20000/600	2	12500

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_D8as_v5	8	32	Remote Storage Only	16	12800/200	20000/600	4	12500
Standard_D16as_v5	16	64	Remote Storage Only	32	25600/384	40000/800	8	12500
Standard_D32as_v5	32	128	Remote Storage Only	32	51200/768	80000/1600	8	16000
Standard_D48as_v5	48	192	Remote Storage Only	32	76800/1152	80000/2000	8	24000
Standard_D64as_v5	64	256	Remote Storage Only	32	80000/1200	80000/2000	8	32000
Standard_D96as_v5	96	384	Remote Storage Only	32	80000/1600	80000/2000	8	40000

<sup>1</sup> Dasv5-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

## Dadsv5-series

Dadsv5-series utilize AMD's 3rd Generation EPYC™ 7763v processors that can achieve a boosted maximum frequency of 3.5GHz. The Dadsv5-series sizes offer a combination of vCPU, memory and temporary storage for most production workloads. The new VMs have 50% larger local storage, as well as better local disk IOPS for both read and write compared to the [Dav4/Dasv4](#) sizes with [Gen2](#) VMs.

Dadsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORE THROU GPUT: IOPS/M BPS	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS <sup>1</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_D2ads_v5	2	8	75	4	9000 / 125	3750/82	10000/600	2	12500
Standard_D4ads_v5	4	16	150	8	19000 / 250	6400/144	20000/600	2	12500
Standard_D8ads_v5	8	32	300	16	38000 / 500	12800/200	20000/600	4	12500
Standard_D16ads_v5	16	64	600	32	75000 / 1000	25600/384	40000/800	8	12500
Standard_D32ads_v5	32	128	1200	32	150000 / 2000	51200/768	80000/1000	8	16000
Standard_D48ads_v5	48	192	1800	32	225000 / 3000	76800/1152	80000/2000	8	24000
Standard_D64ads_v5	64	256	2400	32	300000 / 4000	80000/1200	80000/2000	8	32000
Standard_D96ads_v5	96	384	3600	32	450000 / 4000	80000/1600	80000/2000	8	40000

- These IOPs values can be achieved by using Gen2 VMs.

<sup>1</sup> Dadsv5-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

## Size table definitions

- Storage capacity is shown in units of GiB or 1024<sup>3</sup> bytes. When you compare disks measured in GB (1000<sup>3</sup> bytes) to disks measured in GiB (1024<sup>3</sup>) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps = 10<sup>6</sup> bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk](#)

performance.

- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# DCasv5 and DCadsv5-series confidential VMs

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

The DCasv5-series and DCadsv5-series are [confidential VMs](#) for use in Confidential Computing.

These confidential VMs use AMD's third-Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache. These processors can achieve a boosted maximum frequency of 3.5 GHz. Both series offer Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP). SEV-SNP provides hardware-isolated VMs that protect data from other VMs, the hypervisor, and host management code. Confidential VMs offer hardware-based VM memory encryption. These series also offer OS disk pre-encryption before VM provisioning with different key management solutions.

## DCasv5-series

DCasv5-series VMs offer a combination of vCPU and memory for most production workloads. These VMs with no local disk provide a better value proposition for workloads where you don't need a local temporary disk. For more information, see the [FAQ for Azure VM sizes with no local temporary disk](#).

This series supports Standard SSD, Standard HDD, and Premium SSD disk types. Billing for disk storage and VMs is separate. To estimate your costs, use the [Pricing Calculator](#).

### NOTE

There are some [pricing differences based on your encryption settings](#) for confidential VMs.

### DCasv5-series feature support

*Supported* features in DCasv5-series VMs:

- [Premium Storage](#)
- [Premium Storage caching](#)
- [VM Generation 2](#)

*Unsupported* features in DCasv5-series VMs:

- [Live Migration](#)
- [Memory Preserving Updates](#)
- [Accelerated Networking](#)
- [Ephemeral OS Disks](#)
- [Nested Virtualization](#)

### DCasv5-series products

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_DC2as_v5	2	8	Remote Storage Only	4	3750/82	2

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_DC_4as_v5	4	16	Remote Storage Only	8	6400/144	2
Standard_DC_8as_v5	8	32	Remote Storage Only	16	12800/200	4
Standard_DC_16as_v5	16	64	Remote Storage Only	32	25600/384	4
Standard_DC_32as_v5	32	128	Remote Storage Only	32	51200/768	8
Standard_DC_48as_v5	48	192	Remote Storage Only	32	76800/1152	8
Standard_DC_64as_v5	64	256	Remote Storage Only	32	80000/1200	8
Standard_DC_96as_v5	96	384	Remote Storage Only	32	80000/1600	8

## DCadsV5-series

DCadsV5-series offer a combination of vCPU, memory, and temporary storage for most production workloads.

This series supports Standard SSD, Standard HDD, and Premium SSD disk types. Billing for disk storage and VMs is separate. To estimate your costs, use the [Pricing Calculator](#).

### NOTE

There are some [pricing differences based on your encryption settings](#) for confidential VMs.

### DCadsV5-series feature support

*Supported* features in DCadsV5-series VMs:

- [Premium Storage](#)
- [Premium Storage caching](#)
- [VM Generation 2](#)

*Unsupported* features in DCadsV5-series VMs:

- [Live Migration](#)
- [Memory Preserving Updates](#)
- [Accelerated Networking](#)
- [Ephemeral OS Disks](#)

### DCadsV5-series products

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_D C2ads_v5	2	8	75	4	9000 / 125	3750/82	2
Standard_D C4ads_v5	4	16	150	8	19000 / 250	6400/144	2
Standard_D C8ads_v5	8	32	300	16	38000 / 500	12800/200	4
Standard_D C16ads_v5	16	64	600	32	75000 / 1000	25600/384	4
Standard_D C32ads_v5	32	128	1200	32	150000 / 2000	51200/768	8
Standard_D C48ads_v5	48	192	1800	32	225000 / 3000	76800/115 2	8
Standard_D C64ads_v5	64	256	2400	32	300000 / 4000	80000/120 0	8
Standard_D C96ads_v5	96	384	3600	32	450000 / 4000	80000/160 0	8

#### NOTE

To achieve these IOPs, use [Gen2 VMs](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize](#)

[network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Next steps

[Confidential virtual machine options on AMD processors](#)

# Dpsv5 and Dpdsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dpsv5-series and Dpdsv5-series virtual machines are based on the Arm architecture, delivering outstanding price-performance for general-purpose workloads. These virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer a range of vCPU sizes, up to 4 GiB of memory per vCPU, and temporary storage options able to meet the requirements of scale-out and most enterprise workloads such as web and application servers, small to medium databases, caches, and more.

## Dpsv5-series

Dpsv5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 64 vCPU and 208 GiB of RAM and are optimized for scale-out and most enterprise workloads. Dpsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types with no local-SSD support. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Not supported
- [Nested Virtualization](#): Not supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWORK BANDWID TH (MBPS)
Standard_D2ps_v5	2	8	Remote Storage Only	4	3750/85	10000/1200	2	12500
Standard_D4ps_v5	4	16	Remote Storage Only	8	6400/145	20000/1200	2	12500
Standard_D8ps_v5	8	32	Remote Storage Only	16	12800/290	20000/1200	4	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_D16ps_v5	16	64	Remote Storage Only	32	25600/600	40000/1200	4	12500
Standard_D32ps_v5	32	128	Remote Storage Only	32	51200/865	80000/2000	8	16000
Standard_D48ps_v5	48	192	Remote Storage Only	32	76800/1315	80000/3000	8	24000
Standard_D64ps_v5	64	208	Remote Storage Only	32	80000/1735	80000/3000	8	40000

#### NOTE

Accelerated networking is required and turned on by default on all Dpsv5 machines.

## Dpdsv5-series

Dpdsv5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 64 vCPU, 208 GiB of RAM, and fast local SSD storage with up to 2,400 GiB in capacity and are optimized for scale-out and most enterprise workloads. Dpdsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- [Nested Virtualization](#): Not supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX NICs	MAX NETWORK BANDWIDTH (MBPS)
Standard_D2pd_s_v5	2	8	75	4	9375/125	3750/85	10000/1200	2	12500
Standard_D4pd_s_v5	4	16	150	8	19000/250	6400/145	20000/1200	2	12500
Standard_D8pd_s_v5	8	32	300	16	38000/500	12800/290	20000/1200	4	12500
Standard_D16pds_v5	16	64	600	32	75000/1000	25600/600	40000/1200	4	12500
Standard_D32pds_v5	32	128	1200	32	150000/2000	51200/865	80000/2000	8	16000
Standard_D48pds_v5	48	192	1800	32	225000/3000	76800/1315	80000/3000	8	24000
Standard_D64pds_v5	64	208	2400	32	300000/4000	80000/1735	80000/3000	8	40000

#### NOTE

Accelerated networking is required and turned on by default on all Dpsv5 machines.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all

NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Dplsv5 and Dpldsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Dplsv5-series and Dpldsv5-series virtual machines are based on the Arm architecture, delivering outstanding price-performance for general-purpose workloads. These virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer a range of vCPU sizes, up to 2 GiB of memory per vCPU, and temporary storage options able to meet the requirements of most non-memory-intensive and scale-out workloads such as microservices, small databases, caches, gaming servers, and more.

## Dplsv5-series

Dplsv5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 64 vCPU and 128 GiB of RAM and offer a better value proposition for non-memory-intensive scale-out workloads. Dplsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types with no local-SSD support. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Not supported
- [Nested Virtualization](#): Not supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_D2pls_v5	2	4	Remote Storage Only	4	3750/85	10000/12 00	2	12500
Standard_D4pls_v5	4	8	Remote Storage Only	8	6400/145	20000/12 00	2	12500
Standard_D8pls_v5	8	16	Remote Storage Only	16	12800/29 0	20000/12 00	4	12500

SIZE	vCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_D16pls_v5	16	32	Remote Storage Only	32	25600/600	40000/1200	4	12500
Standard_D32pls_v5	32	64	Remote Storage Only	32	51200/865	80000/2000	8	16000
Standard_D48pls_v5	48	96	Remote Storage Only	32	76800/1315	80000/3000	8	24000
Standard_D64pls_v5	64	128	Remote Storage Only	32	80000/1735	80000/3000	8	40000

#### NOTE

Accelerated networking is required and turned on by default on all Dplsv5 machines.

## Dpldsv5-series

Dpldsv5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 64 vCPU, 128 GiB of RAM, and fast local SSD storage up to 2,400 GiB built for scale-out, non-memory-intensive workloads that require local disk. Dpldsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- [Nested Virtualization](#): Not supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX NICs	MAX NETWORK BANDWIDTH (MBPS)
Standard_D2plds_v5	2	4	75	4	9375/125	3750/85	10000/1200	2	12500
Standard_D4plds_v5	4	8	150	8	19000/250	6400/145	20000/1200	2	12500
Standard_D8plds_v5	8	16	300	16	38000/500	12800/290	20000/1200	4	12500
Standard_D16plds_v5	16	32	600	32	75000/1000	25600/600	40000/1200	4	12500
Standard_D32plds_v5	32	64	1200	32	150000/2000	51200/865	80000/2000	8	16000
Standard_D48plds_v5	48	96	1800	32	225000/3000	76800/1315	80000/3000	8	24000
Standard_D64plds_v5	64	128	2400	32	300000/4000	80000/1735	80000/3000	8	40000

#### NOTE

Accelerated networking is required and turned on by default on all Dplsv5 machines.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all

NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Compute optimized virtual machine sizes

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

Compute optimized VM sizes have a high CPU-to-memory ratio. These sizes are good for medium traffic web servers, network appliances, batch processes, and application servers. This article provides information about the number of vCPUs, data disks, and NICs. It also includes information about storage throughput and network bandwidth for each size in this grouping.

- The [Fsv2-series](#) runs on 2nd Generation Intel® Xeon® Platinum 8272CL (Cascade Lake) processors and Intel® Xeon® Platinum 8168 (Skylake) processors. It features a sustained all core Turbo clock speed of 3.4 GHz and a maximum single-core turbo frequency of 3.7 GHz. Intel® AVX-512 instructions are new on Intel Scalable Processors. These instructions provide up to a 2X performance boost to vector processing workloads on both single and double precision floating point operations. In other words, they're really fast for any computational workload. At a lower per-hour list price, the Fsv2-series is the best value in price-performance in the Azure portfolio based on the Azure Compute Unit (ACU) per vCPU.
- The [FX-series](#) runs on the Intel® Xeon® Gold 6246R (Cascade Lake) processors. It features an all-core-turbo frequency of 4.0GHz, 21GB RAM per vCPU, up to 1TB total RAM, and local temporary storage. It will benefit workloads which require a high CPU clock speed and high memory to CPU ratio, workloads with high per-core licensing costs, and applications requiring high a single-core performance. A typical use case for FX-series is the Electronic Design Automation (EDA) workload.

## Other sizes

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

For more information on how Azure names its VMs, see [Azure virtual machine sizes naming conventions](#).

# Fsv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Fsv2-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors, or the Intel® Xeon® Platinum 8168 (Skylake) processors. It features a sustained all core Turbo clock speed of 3.4 GHz and a maximum single-core turbo frequency of 3.7 GHz. Intel® AVX-512 instructions are new on Intel Scalable Processors. These instructions provide up to a 2X performance boost to vector processing workloads on both single and double precision floating point operations. In other words, they're really fast for any computational workload.

Fsv2-series VMs feature Intel® Hyper-Threading Technology.

**ACU:** 195 - 210

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Supported

SIZE	VCPU'S	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS (CACHE SIZE IN GIB)	MAX UNCACHED DISK THROUGHPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROUGHPUT: IOPS/M BPS <sup>1</sup>	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_F2s_v2 <sup>4</sup>	2	4	16	4	4000/31 (32)	3200/47	4000/200	2	5000
Standard_F4s_v2	4	8	32	8	8000/63 (64)	6400/95	8000/200	2	10000
Standard_F8s_v2	8	16	64	16	16000/127 (128)	12800/190	16000/400	4	12500
Standard_F16s_v2	16	32	128	32	32000/255 (256)	25600/380	32000/800	4	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS (CACHE SIZE IN GiB)	MAX UNCACLED DISK THROUGHPUT: IOPS/M BPS	MAX BURST UNCACLED DISK THROUGHPUT: IOPS/M BPS	MAX NICS	EXPECT ED NETWORK BANDWIDTH (MBPS)
Standard_F32s_v2	32	64	256	32	64000/512 (512)	51200/750	64000/1600	8	16000
Standard_F48s_v2	48	96	384	32	96000/768 (768)	76800/1100	80000/2000	8	21000
Standard_F64s_v2	64	128	512	32	128000/1024 (1024)	80000/1100	80000/2000	8	28000
Standard_F72s_v2 <sup>2, 3</sup>	72	144	576	32	144000/1152 (1520)	80000/1100	80000/2000	8	30000

<sup>1</sup> Fsv2-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> The use of more than 64 vCPU require one of these supported guest operating systems:

- Windows Server 2016 or later
- Ubuntu 16.04 LTS or later, with Azure tuned kernel (4.15 kernel or later)
- SLES 12 SP2 or later
- RHEL or CentOS version 6.7 through 6.10, with Microsoft-provided LIS package 4.3.1 (or later) installed
- RHEL or CentOS version 7.3, with Microsoft-provided LIS package 4.2.1 (or later) installed
- RHEL or CentOS version 7.6 or later
- Oracle Linux with UEK4 or later
- Debian 9 with the backports kernel, Debian 10 or later
- CoreOS with a 4.14 kernel or later

<sup>3</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>4</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or 1024^3 bytes. When you compare disks measured in GB (1000^3 bytes) to disks measured in GiB (1024^3) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps = 10^6 bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to

## None.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# FX-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

The FX-series runs on the Intel® Xeon® Gold 6246R (Cascade Lake) processors. It features an all-core-turbo frequency of 4.0 GHz, 21 GB RAM per vCPU, up to 1 TB total RAM, and local temporary storage. The FX-series will benefit workloads that require a high CPU clock speed and high memory to CPU ratio, workloads with high per-core licensing costs, and applications requiring a high single-core performance. A typical use case for FX-series is the Electronic Design Automation (EDA) workload.

FX-series VMs feature Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology, and Intel® Advanced Vector Extensions 512 (Intel® AVX-512).

**ACU:** 310 - 340

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_FX4mds	4	84	168	8	40000/343	6700/104	2	4000
Standard_FX12mds	12	252	504	24	100000/1029	20000/314	4	8000
Standard_FX24mds	24	504	1008	32	200000/2057	40000/629	4	16000
Standard_FX36mds	36	756	1512	32	300000/3086	60000/944	8	24000
Standard_FX48mds	48	1008	2016	32	400000/3871	80000/1258	8	32000

## Size table definitions

- Storage capacity is shown in units of GiB or 1024^3 bytes. When you compare disks measured in GB (1000^3 bytes) to disks measured in GiB (1024^3) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Memory optimized virtual machine sizes

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Memory optimized VM sizes offer a high memory-to-CPU ratio that is great for relational database servers, medium to large caches, and in-memory analytics. This article provides information about the number of vCPUs, data disks and NICs. You can also learn about storage throughput and network bandwidth for each size in this grouping.

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

- **Dv2 and DSv2-series**, a follow-on to the original D-series, features a more powerful CPU. The Dv2-series is about 35% faster than the D-series. It runs on the Intel® Xeon® 8171M 2.1 GHz (Skylake) or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell) or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors, and with the Intel Turbo Boost Technology 2.0. The Dv2-series has the same memory and disk configurations as the D-series.

Dv2 and DSv2-series are ideal for applications that demand faster vCPUs, better temporary storage performance, or have higher memory demands. They offer a powerful combination for many enterprise-grade applications.

- The **Eav4 and Easv4-series** utilize AMD's 2.35Ghz EPYC™ 7452 processor in a multi-threaded configuration with up to 256 MB L3 cache, increasing options for running most memory optimized workloads. The Eav4-series and Easv4-series have the same memory and disk configurations as the Ev4 & Esv3-series.
- The **Ebsv5 and Ebdsv5 series** deliver higher remote storage performance in each VM size than the Ev4 series. The increased remote storage performance of the Ebsv5 and Ebdsv5 VMs is ideal for storage throughput-intensive workloads, such as relational databases and data analytics applications.
- The **Ev3 and Esv3-series** feature the Intel® Xeon® 8171M 2.1 GHz (Skylake) or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell) processor in a hyper-threaded configuration. This configuration provides a better value proposition for most general purpose workloads, and brings the Ev3 into alignment with the general purpose VMs of most other clouds. Memory has been expanded (from 7 GiB/vCPU to 8 GiB/vCPU) while disk and network limits have been adjusted on a per core basis to align with the move to hyper-threading. The Ev3 is the follow up to the high memory VM sizes of the D/Dv2 families.
- The **Ev4 and Esv4-series** runs on 2nd Generation Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, are ideal for various memory-intensive enterprise applications and feature up to 504 GiB of RAM. It features the [Intel® Turbo Boost Technology 2.0](#), [Intel® Hyper-Threading Technology](#) and [Intel® Advanced Vector Extensions 512 \(Intel AVX-512\)](#). The Ev4 and Esv4-series don't include a local temp disk. For more information, see [Azure VM sizes with no local temp disk](#).
- The **Edv4 and Edsv4-series** runs on 2nd Generation Intel® Xeon® Platinum 8272CL (Cascade Lake) processors, ideal for extremely large databases or other applications that benefit from high vCPU counts and large amounts of memory. Additionally, these VM sizes include fast, larger local SSD storage for applications that benefit from low latency, high-speed local storage. It features an all core Turbo clock

speed of 3.4 GHz, Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology and Intel® Advanced Vector Extensions 512 (Intel AVX-512).

- The [Easv5 and Eadsv5-series](#) utilize AMD's 3rd Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache, increasing customer options for running most memory optimized workloads. These virtual machines offer a combination of vCPUs and memory to meet the requirements associated with most memory-intensive enterprise applications, such as relational database servers and in-memory analytics workloads.
- The [Edv5 and Edsv5-series](#) run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processors in a hyper-threaded configuration. These series are ideal for various memory-intensive enterprise applications. They feature up to 672 GiB of RAM, Intel® Turbo Boost Technology 2.0, Intel® Hyper-Threading Technology and Intel® Advanced Vector Extensions 512 (Intel® AVX-512). The series also support Intel® Deep Learning Boost. These new VM sizes have 50% larger local storage, and better local disk IOPS for both read and write compared to the [Ev3/Esv3](#) sizes with [Gen2 VMs](#). It features an all core Turbo clock speed of 3.4 GHz.
- The [Epsv5 and Epdsv5-series](#) are ARM64-based VMs featuring the 80 core, 3.0 GHz Ampere Altra processor. These series are designed for common enterprise workloads. They're optimized for database, in-memory caching, analytics, gaming, web, and application servers running on Linux.
- The [Ev5 and Esv5-series](#) runs on the Intel® Xeon® Platinum 8272CL (Ice Lake) processors in a hyper-threaded configuration, are ideal for various memory-intensive enterprise applications and feature up to 512 GiB of RAM. It features an all core Turbo clock speed of 3.4 GHz.
- The [M-series](#) offers a high vCPU count (up to 128 vCPUs) and a large amount of memory (up to 3.8 TiB). It's also ideal for extremely large databases or other applications that benefit from high vCPU counts and large amounts of memory.
- The [Mv2-series](#) offers the highest vCPU count (up to 416 vCPUs) and largest memory (up to 11.4 TiB) of any VM in the cloud. It's ideal for extremely large databases or other applications that benefit from high vCPU counts and large amounts of memory.

Azure Compute offers virtual machine sizes that are isolated to a specific hardware type and dedicated to a single customer. These virtual machine sizes are best suited for workloads that require a high degree of isolation from other customers for workloads involving elements like compliance and regulatory requirements. Customers can also choose to further subdivide the resources of these isolated virtual machines by using [Azure support for nested virtual machines](#). See the pages for virtual machine families below for your isolated VM options.

## Other sizes

- [General purpose](#)
- [Compute optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

For more information on how Azure names its VMs, see [Azure virtual machine sizes naming conventions](#).

# Memory optimized Dv2 and Dsv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Dv2 and Dsv2-series, a follow-on to the original D-series, features a more powerful CPU. DSv2-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake), the Intel® Xeon® 8171M 2.1 GHz (Skylake), the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors. The Dv2-series has the same memory and disk configurations as the D-series.

## Dv2-series 11-15

Dv2-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake), the Intel® Xeon® 8171M 2.1 GHz (Skylake), the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors.

**ACU:** 210 - 250

**Premium Storage:** Not Supported

**Premium Storage caching:** Not Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Not Supported

**Nested Virtualization:** Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THROUGHPUT: IOPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D 11_v2	2	14	100	6000/93/46	8/8x500	2	1500
Standard_D 12_v2	4	28	200	12000/187/93	16/16x500	4	3000
Standard_D 13_v2	8	56	400	24000/375/187	32/32x500	8	6000
Standard_D 14_v2	16	112	800	48000/750/375	64/64x500	8	12000
Standard_D 15_v2 <sup>1</sup>	20	140	1000	60000/937/468	64/64x500	8	25000 <sup>2</sup>

<sup>1</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>2</sup> 25000 Mbps with Accelerated Networking.

## DSv2-series 11-15

DSv2-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), the Intel® Xeon® Platinum 8272CL (Cascade Lake), the Intel® Xeon® 8171M 2.1 GHz (Skylake), the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell), or the Intel® Xeon® E5-2673 v3 2.4 GHz (Haswell) processors.

[ACU](#): 210 - 250<sup>1</sup>

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUG HPUT: IOPS/MB PS (CACHE SIZE IN GiB)	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	EXPECTE D NETWOR K BANDWID TH (Mbps)
Standard_DS11_v2 <sup>3</sup>	2	14	28	8	8000/64 (72)	6400/96	2	1500
Standard_DS12_v2 <sup>3</sup>	4	28	56	16	16000/128 (144)	12800/192	4	3000
Standard_DS13_v2 <sup>3</sup>	8	56	112	32	32000/256 (288)	25600/384	8	6000
Standard_DS14_v2 <sup>3</sup>	16	112	224	64	64000/512 (576)	51200/768	8	12000
Standard_DS15_v2 <sup>2</sup>	20	140	280	64	80000/640 (720)	64000/960	8	25000 <sup>4</sup>

<sup>1</sup> The maximum disk throughput (IOPS or MBps) possible with a DSv2 series VM may be limited by the number, size and striping of the attached disk(s). For details, see [Designing for high performance](#).<sup>2</sup> Instance is isolated to the Intel Haswell based hardware and dedicated to a single customer.

<sup>3</sup> Constrained core sizes available.

<sup>4</sup> 25000 Mbps with Accelerated Networking.

## Size table definitions

- Storage capacity is shown in units of GiB or 1024<sup>3</sup> bytes. When you compare disks measured in GB

( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Ev3 and Esv3-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Ev3 and Esv3-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1 GHz (Skylake), or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell) processor in a hyper-threaded configuration, providing a better value proposition for most general purpose workloads, and bringing the Ev3 into alignment with the general purpose VMs of most other clouds. Memory has been expanded (from 7 GiB/vCPU to 8 GiB/vCPU) while disk and network limits have been adjusted on a per core basis to align with the move to hyperthreading. The Ev3 is the follow up to the high memory VM sizes of the D/Dv2 families.

## Ev3-series

Ev3-series instances run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1 GHz (Skylake), or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell) processors, and feature Intel Turbo Boost Technology 2.0. Ev3-series instances are ideal for memory-intensive enterprise applications.

Data disk storage is billed separately from virtual machines. To use premium storage disks, use the ESv3 sizes. The pricing and billing meters for ESv3 sizes are the same as Ev3-series.

Ev3-series VM's feature Intel® Hyper-Threading Technology.

[ACU](#): 160 - 190

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICs / NETWORK BANDWIDTH
Standard_E2_v3 <sup>1</sup>	2	16	50	4	3000/46/23	2/1000
Standard_E4_v3	4	32	100	8	6000/93/46	2/2000
Standard_E8_v3	8	64	200	16	12000/187/93	4/4000

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICs / NETWORK BANDWIDTH
Standard_E16_v3	16	128	400	32	24000/375/187	8/8000
Standard_E20_v3	20	160	500	32	30000/469/234	8/10000
Standard_E32_v3	32	256	800	32	48000/750/375	8/16000
Standard_E48_v3	48	384	1200	32	96000/1000/500	8/24000
Standard_E64_v3	64	432	1600	32	96000/1000/500	8/30000
Standard_E64i_v3 <sup>2</sup>	64	432	1600	32	96000/1000/500	8/30000

<sup>1</sup> Accelerated networking can only be applied to a single NIC. <sup>2</sup> Instance is isolated to hardware dedicated to a single customer.

## Esv3-series

Esv3-series instances run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake), Intel® Xeon® Platinum 8272CL (Cascade Lake), Intel® Xeon® 8171M 2.1 GHz (Skylake), or the Intel® Xeon® E5-2673 v4 2.3 GHz (Broadwell) processor, feature Intel Turbo Boost Technology 2.0 and use premium storage. Esv3-series instances are ideal for memory-intensive enterprise applications.

Esv3-series VM's feature Intel® Hyper-Threading Technology.

[ACU](#): 160-190

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS (CACHE SIZE IN GiB)	BURST CACHE D AND TEMP STORAGE THROUGHPUT: IOPS/M BPS <sup>3</sup>	MAX UNCACHED DISK THROUGHPUT: IOPS/M BPS	BURST UNCACHED DISK THROUGHPUT: IOPS/M BPS <sup>3</sup>	MAX NICS/EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_E2s_v3 <sup>4</sup>	2	16	32	4	4000/32 (50)	4000/100	3200/48	4000/200	2/1000
Standard_E4s_v3 <sup>1</sup>	4	32	64	8	8000/64 (100)	8000/200	6400/96	8000/200	2/2000
Standard_E8s_v3 <sup>1</sup>	8	64	128	16	16000/128 (200)	16000/400	12800/192	16000/400	4/4000
Standard_E16s_v3 <sup>1</sup>	16	128	256	32	32000/256 (400)	32000/800	25600/384	32000/800	8/8000
Standard_E20s_v3	20	160	320	32	40000/320 (400)	40000/1000	32000/480	40000/1000	8/10000
Standard_E32s_v3 <sup>1</sup>	32	256	512	32	64000/512 (800)	64000/1600	51200/768	64000/1600	8/16000
Standard_E48s_v3	48	384	768	32	96000/768 (1200)	96000/2000	76800/1152	80000/2000	8/24000
Standard_E64s_v3 <sup>1</sup>	64	432	864	32	128000/1024 (1600)	128000/2000	80000/1200	80000/2000	8/30000
Standard_E64is_v3 <sup>2</sup>	64	432	864	32	128000/1024 (1600)	128000/2000	80000/1200	80000/2000	8/30000

<sup>1</sup> Constrained core sizes available.

<sup>2</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>3</sup> Esv3-series VMs can burst their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>4</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may

appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Eav4 and Easv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Eav4-series and Easv4-series run on 2nd Generation AMD EPYC™ 7452 or 3rd Generation EPYC™ 7763v processors in a multi-threaded configuration. The Eav4-series and Easv4-series have the same memory and disk configurations as the Ev3 & Esv3-series.

## Eav4-series

[ACU](#): 230 - 260

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generations 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

The Eav4-series run on 2nd Generation AMD EPYC™ 7452 (up to 3.35GHz) or 3rd Generation EPYC™ 7763v processors (up to 3.5GHz). The Eav4-series sizes are ideal for memory-intensive enterprise applications. Data disk storage is billed separately from virtual machines. To use premium SSD, use the Easv4-series sizes. The pricing and billing meters for Easv4 sizes are the same as the Eav3-series.

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_E_2a_v4 <sup>1</sup>	2	16	50	4	3000 / 46 / 23	2	800
Standard_E_4a_v4	4	32	100	8	6000 / 93 / 46	2	1600
Standard_E_8a_v4	8	64	200	16	12000 / 187 / 93	4	3200
Standard_E_16a_v4	16	128	400	32	24000 / 375 / 187	8	6400
Standard_E_20a_v4	20	160	500	32	30000 / 468 / 234	8	8000
Standard_E_32a_v4	32	256	800	32	48000 / 750 / 375	8	12800

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS / READ MBPS / WRITE MBPS	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_E 48a_v4	48	384	1200	32	96000 / 1000 (500)	8	19200
Standard_E 64a_v4	64	512	1600	32	96000 / 1000 (500)	8	25600
Standard_E 96a_v4	96	672	2400	32	96000 / 1000 (500)	8	32000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Easv4-series

[ACU](#): 230 - 260

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generations 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

The Easv4-series run on 2nd Generation AMD EPYC™ 7452 (up to 3.35GHz) or 3rd Generation EPYC™ 7763v processors (up to 3.5GHz) and use premium SSD. The Easv4-series sizes are ideal for memory-intensive enterprise applications.

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORE GE THROU GHPUT : IOPS / MBPS (CACHE SIZE IN GIB)	MAX BURST CACHE D AND TEMP STORE GE THROU GHPUT : IOPS / MBPS <sup>1</sup>	MAX UNCA CHED DISK THROU GHPUT : IOPS / MBPS	MAX BURST UNCA CHED DISK THROU GHPUT : IOPS/ MBPS <sup>1</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_E2a_s_v4 <sup>3</sup>	2	16	32	4	4000 / 32 (50)	4000/100	3200 / 48	4000/200	2	800
Standard_E4a_s_v4 <sup>2</sup>	4	32	64	8	8000 / 64 (100)	8000/200	6400 / 96	8000/200	2	1600

SIZE	VCPUs	MEMORY: GIB	TEMP STORE GE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORA GE THROU GHPUT : IOPS / MBPS (CACH E SIZE IN GIB)	MAX BURST CACHE D AND TEMP STORA GE THROU GHPUT : IOPS / MBPS	MAX UNCA CHED DISK THROU GHPUT : IOPS / MBPS	MAX BURST UNCA CHED DISK THROU GHPUT : IOPS/ MBPS	MAX NICS	EXPEC TED NETW ORK BAND WIDTH (MBPS)
Standard_E8as_v4 <sup>2</sup>	8	64	128	16	16000 / 128 (200)	16000/ 400	12800 / 192	16000/ 400	4	3200
Standard_E16as_v4 <sup>2</sup>	16	128	256	32	32000 / 255 (400)	32000/ 800	25600 / 384	32000/ 800	8	6400
Standard_E20as_v4	20	160	320	32	40000 / 320 (500)	40000/ 1000	32000 / 480	40000/ 1000	8	8000
Standard_E32as_v4 <sup>2</sup>	32	256	512	32	64000 / 510 (800)	64000/ 1600	51200 / 768	64000/ 1600	8	12800
Standard_E48as_v4	48	384	768	32	96000 / 1020 (1200)	96000/ 2000	76800 / 1148	80000/ 2000	8	19200
Standard_E64as_v4 <sup>2</sup>	64	512	1024	32	12800 0 / 1020 (1600)	12800 0/2000	80000 / 1200	80000/ 2000	8	25600
Standard_E96as_v4 <sup>2</sup>	96	672	1344	32	19200 0 / 1020 (2400)	19200 0/2000	80000 / 1200	80000/ 2000	8	32000

<sup>1</sup> Easv4-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> [Constrained core sizes available.](#)

<sup>3</sup> Accelerated networking can only be applied to a single NIC.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Edv4 and Edsv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Edv4 and Edsv4-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, and are ideal for various memory-intensive enterprise applications and feature up to 504 GiB of RAM, [Intel® Turbo Boost Technology 2.0](#), [Intel® Hyper-Threading Technology](#) and [Intel® Advanced Vector Extensions 512 \(Intel® AVX-512\)](#). They also support [Intel® Deep Learning Boost](#). These new VM sizes will have 50% larger local storage, as well as better local disk IOPS for both read and write compared to the [Ev3/Esv3](#) sizes with [Gen2](#) VMs. It features an all core Turbo clock speed of 3.4 GHz.

## Edv4-series

Edv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors. The Edv4 virtual machine sizes feature up to 504 GiB of RAM, in addition to fast and large local SSD storage (up to 2,400 GiB). These virtual machines are ideal for memory-intensive enterprise applications and applications that benefit from low latency, high-speed local storage. You can attach Standard SSDs and Standard HDDs disk storage to the Edv4 VMs.

[ACU](#): 195 - 210

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported<sup>1</sup>

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/Mbps *	MAX NICS	MAX NETWORK BANDWIDT H (Mbps)
Standard_E_2d_v4 <sup>1</sup>	2	16	75	4	9000/125	2	5000
Standard_E_4d_v4	4	32	150	8	19000/250	2	10000
Standard_E_8d_v4	8	64	300	16	38000/500	4	12500
Standard_E_16d_v4	16	128	600	32	75000/1000	8	12500

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUHPUT: IOPS/MBPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E 20d_v4	20	160	750	32	94000/1250	8	16000
Standard_E 32d_v4	32	256	1200	32	150000/2000	8	16000
Standard_E 48d_v4	48	384	1800	32	225000/3000	8	24000
Standard_E 64d_v4	64	504	2400	32	300000/4000	8	30000

\* These IOPs values can be achieved by using [Gen2 VMs](#) <sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Edsv4-series

Edsv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors. The Edsv4 virtual machine sizes feature up to 504 GiB of RAM, in addition to fast and large local SSD storage (up to 2,400 GiB). These virtual machines are ideal for memory-intensive enterprise applications and applications that benefit from low latency, high-speed local storage.

**ACU:** 195-210

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUHPUT: IOPS/M BPS*	MAX UNCAC HED DISK THROU GHPUT: IOPS/M BPS	MAX UNCAC HED DISK THROU GHPUT: IOPS/M BPS <sup>1</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E2ds_v4 <sup>4</sup>	2	16	75	4	9000/125	3200/48	4000/200	2	5000
Standard_E4ds_v4	4	32	150	8	19000/250	6400/96	8000/200	2	10000

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E8ds_v4	8	64	300	16	38000/500	12800/192	16000/400	4	12500
Standard_E16ds_v4	16	128	600	32	75000/1000	25600/384	32000/800	8	12500
Standard_E20ds_v4	20	160	750	32	94000/1250	32000/480	40000/1000	8	16000
Standard_E32ds_v4	32	256	1200	32	150000/2000	51200/768	64000/1600	8	16000
Standard_E48ds_v4	48	384	1800	32	225000/3000	76800/1152	80000/2000	8	24000
Standard_E64ds_v4 <sup>2</sup>	64	504	2400	32	300000/4000	80000/1200	80000/2000	8	30000
Standard_E80ids_v4 <sup>3,5</sup>	80	504	2400	64	375000/4000	80000/1200	80000/2000	8	30000

\* These IOPs values can be guaranteed by using [Gen2 VMs](#)

<sup>1</sup> Edsv4-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> [Constrained core sizes available.](#)

<sup>3</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>4</sup> Accelerated networking can only be applied to a single NIC.

<sup>5</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_E80ids\_v4** results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 120000/1800
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 120000/2000

## Size table definitions

- Storage capacity is shown in units of GiB or 1024^3 bytes. When you compare disks measured in GB (1000^3 bytes) to disks measured in GiB (1024^3) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Ev4 and Esv4-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Ev4 and Esv4-series run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake) processors in a hyper-threaded configuration, are ideal for various memory-intensive enterprise applications and feature up to 504GiB of RAM. It features an all core Turbo clock speed of 3.4 GHz.

## NOTE

For frequently asked questions, refer to [Azure VM sizes with no local temp disk](#).

## Ev4-series

Ev4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel Xeon® Platinum 8272CL (Cascade Lake). The Ev4-series instances are ideal for memory-intensive enterprise applications. Ev4-series VMs feature Intel® Hyper-Threading Technology.

Remote Data disk storage is billed separately from virtual machines. To use premium storage disks, use the Esv4 sizes. The pricing and billing meters for Esv4 sizes are the same as Ev4-series.

**ACU:** 195 - 210

**Premium Storage:** Not Supported

**Premium Storage caching:** Not Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Not Supported

**Nested Virtualization:** Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_E2_v4 <sup>1</sup>	2	16	Remote Storage Only	4	2	5000
Standard_E4_v4	4	32	Remote Storage Only	8	2	10000
Standard_E8_v4	8	64	Remote Storage Only	16	4	12500
Standard_E16_v4	16	128	Remote Storage Only	32	8	12500

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_E20_v4	20	160	Remote Storage Only	32	8	10000
Standard_E32_v4	32	256	Remote Storage Only	32	8	16000
Standard_E48_v4	48	384	Remote Storage Only	32	8	24000
Standard_E64_v4	64	504	Remote Storage Only	32	8	30000

<sup>1</sup> Accelerated networking can only be applied to a single NIC.

## Esv4-series

Esv4-series sizes run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) or the Intel® Xeon® Platinum 8272CL (Cascade Lake). The Esv4-series instances are ideal for memory-intensive enterprise applications. Esv4-series VMs feature Intel® Hyper-Threading Technology. Remote Data disk storage is billed separately from virtual machines.

[ACU](#): 195-210

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS <sup>1</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_E2s_v4 <sup>4</sup>	2	16	Remote Storage Only	4	3200/48	4000/200	2	5000
Standard_E4s_v4	4	32	Remote Storage Only	8	6400/96	8000/200	2	10000
Standard_E8s_v4	8	64	Remote Storage Only	16	12800/192	16000/400	4	12500

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	EXPECTE D NETWOR K BANDWID TH (MBPS)
Standard_E16s_v4	16	128	Remote Storage Only	32	25600/384	32000/800	8	12500
Standard_E20s_v4	20	160	Remote Storage Only	32	32000/480	40000/1000	8	10000
Standard_E32s_v4	32	256	Remote Storage Only	32	51200/768	64000/1600	8	16000
Standard_E48s_v4	48	384	Remote Storage Only	32	76800/1152	80000/2000	8	24000
Standard_E64s_v4 <sup>2</sup>	64	504	Remote Storage Only	32	80000/1200	80000/2000	8	30000
Standard_E80is_v4 <sup>3,5</sup>	80	504	Remote Storage Only	64	80000/1200	80000/2000	8	30000

<sup>1</sup> Esv4-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> [Constrained core sizes available](#)).

<sup>3</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>4</sup> Accelerated networking can only be applied to a single NIC.

<sup>5</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_E80is\_v4** results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 120000/1800
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 120000/2000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Ev5 and Esv5-series

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Ev5 and Esv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper threaded](#) configuration, providing a better value proposition for most general-purpose workloads. This new processor features an all core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#). Featuring up to 672 GiB of RAM, these virtual machines are ideal for memory-intensive enterprise applications, relational database servers, and in-memory analytics workloads. The Ev5 and Esv5-series provide a better value proposition for workloads that don't require local temp disk. For information about similar virtual machines with local disk, see [Edv5 and Edsv5-series VMs](#).

## NOTE

For frequently asked questions, see [Azure VM sizes with no local temp disk](#).

## Ev5-series

Ev5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 104 vCPU and 672 GiB of RAM. Ev5-series virtual machines don't have temporary storage thus lowering the price of entry.

Ev5-series supports Standard SSD and Standard HDD disk types. To use Premium SSD or Ultra Disk storage, select Esv5-series virtual machines. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	MAX NETWORK BANDWIDTH (Mbps)
Standard_E2_v5 <sup>1,2</sup>	2	16	Remote Storage Only	4	2	12500
Standard_E4_v5	4	32	Remote Storage Only	8	2	12500
Standard_E8_v5	8	64	Remote Storage Only	16	4	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX NICs	MAX NETWORK BANDWIDTH (Mbps)
Standard_E16_v5	16	128	Remote Storage Only	32	8	12500
Standard_E20_v5	20	160	Remote Storage Only	32	8	12500
Standard_E32_v5	32	256	Remote Storage Only	32	8	16000
Standard_E48_v5	48	384	Remote Storage Only	32	8	24000
Standard_E64_v5	64	512	Remote Storage Only	32	8	30000
Standard_E96_v5	96	672	Remote Storage Only	32	8	30000
Standard_E10_4i_v5 <sup>3</sup>	104	672	Remote Storage Only	64	8	100000

<sup>1</sup> Accelerated networking is required and turned on by default on all Ev5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> Instance is [isolated](#) to hardware dedicated to a single customer.

## Esv5-series

Esv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 104 vCPU and 672 GiB of RAM. Esv5-series virtual machines don't have temporary storage thus lowering the price of entry.

Esv5-series supports Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS <sup>5</sup>	MAX NICS	MAX NETWOR K BANDWID TH (MBPS)
Standard_E2s_v5 <sup>1,2</sup>	2	16	Remote Storage Only	4	3750/85	10000/1200	2	12500
Standard_E4s_v5	4	32	Remote Storage Only	8	6400/145	20000/1200	2	12500
Standard_E8s_v5	8	64	Remote Storage Only	16	12800/290	20000/1200	4	12500
Standard_E16s_v5	16	128	Remote Storage Only	32	25600/600	40000/1200	8	12500
Standard_E20s_v5	20	160	Remote Storage Only	32	32000/750	64000/1600	8	12500
Standard_E32s_v5	32	256	Remote Storage Only	32	51200/865	80000/2000	8	16000
Standard_E48s_v5	48	384	Remote Storage Only	32	76800/1315	80000/3000	8	24000
Standard_E64s_v5	64	512	Remote Storage Only	32	80000/1735	80000/3000	8	30000
Standard_E96s_v5 <sup>3</sup>	96	672	Remote Storage Only	32	80000/2600	80000/4000	8	35000
Standard_E104is_v5 <sup>4,6</sup>	104	672	Remote Storage Only	64	120000/4000	120000/4000	8	100000

<sup>1</sup> Accelerated networking is required and turned on by default on all Esv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> [Constrained core](#) sizes available.

<sup>4</sup> Instance is [isolated](#) to hardware dedicated to a single customer.

<sup>5</sup> Esv5-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>6</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_E104is\_v5** results in higher IOPs and MBps than

standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 160000/4000
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 160000/4000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

# Ebdsv5 and Ebsv5 series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The memory-optimized Ebsv5 and Ebdsv5 Azure virtual machine (VM) series deliver higher remote storage performance in each VM size than the [Ev4 series](#). The increased remote storage performance of the Ebsv5 and Ebdsv5 VMs is ideal for storage throughput-intensive workloads. For example, relational databases and data analytics applications.

The Ebsv5 and Ebdsv5 VMs offer up to 120000 IOPS and 4000 MBps of remote disk storage throughput. Both series also include up to 512 GiB of RAM. The Ebdsv5 series has local SSD storage up to 2400 GiB. Both series provide a 3X increase in remote storage performance of data-intensive workloads compared to prior VM generations. You can use these series to consolidate existing workloads on fewer VMs or smaller VM sizes while achieving potential cost savings. The Ebdsv5 series comes with a local disk and Ebsv5 is without a local disk. Standard SSDs and Standard HDD disk storage aren't supported in the Ebsv5 series.

The Ebdsv5 and Ebsv5 series run on the Intel® Xeon® Platinum 8370C (Ice Lake) processors in a hyper-threaded configuration. The series are ideal for various memory-intensive enterprise applications. They feature:

- Up to 512 GiB of RAM
- [Intel® Turbo Boost Technology 2.0](#)
- [Intel® Hyper-Threading Technology](#)
- [Intel® Advanced Vector Extensions 512 \(Intel® AVX-512\)](#)
- Support for [Intel® Deep Learning Boost](#)

## IMPORTANT

- Accelerated networking is required and turned on by default on all Ebsv5 and Ebdsv5 VMs.
- Accelerated networking can be applied to two NICs.
- Ebsv5 and Ebdsv5-series VMs can [burst their disk performance](#) and get up to their bursting max for up to 30 minutes at a time.

## Ebdsv5 series

Ebdsv5-series sizes run on the Intel® Xeon® Platinum 8370C (Ice Lake) processors. The Ebdsv5 VM sizes feature up to 512 GiB of RAM, in addition to fast and large local SSD storage (up to 2400 GiB). These VMs are ideal for memory-intensive enterprise applications and applications that benefit from high remote storage performance, low latency, high-speed local storage. Remote Data disk storage is billed separately from VMs.

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 1 and Generation 2
- [Accelerated Networking](#): Supported (required)
- [Ephemeral OS Disks](#): Supported
- Nested virtualization: Supported

SIZE	VCPU	MEM ORY: GIB	TEMP STOR AGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STOR AGE THRO UGHP UT: IOPS / MBPS	MAX UNCA CHED PREMI UM V1 SSD AND STAN DARD SSD/H DD DISK THRO UGHP UT: IOPS/ MBPS	MAX BURS T UNCA CHED PREMI UM V1 SSD AND STAN DARD SSD/H DD DISK THRO UGHP UT: IOPS/ MBPS	MAX UNCA CHED PREMI UM A DISK AND STAN DARD SSD/H DD DISK THRO UGHP UT: IOPS/ MBPS	MAX BURS T UNCA CHED ULTR A DISK AND PREMI UM V2 SSD DISK THRO UGHP UT: IOPS/ MBPS	MAX NICS	NETW ORK BAND WIDT H
Standard_E_2bds_v5	2	16	75	4	9000/125	5500/156	1000/0/1200	7370/156	1500/0/1200	2	12500
Standard_E_4bds_v5	4	32	150	8	1900/0/250	1100/0/350	2000/0/1200	1474/0/350	3000/0/1200	2	12500
Standard_E_8bds_v5	8	64	300	16	3800/0/500	2200/0/625	4000/0/1200	2948/0/625	6000/0/1200	4	12500
Standard_E_16bds_v5	16	128	600	32	7500/0/1000	4400/0/1250	6400/0/2000	5896/0/1250	9600/0/2000	8	12500
Standard_E_32bds_v5	32	256	1200	32	1500/00/1250	8800/0/2500	1200/00/4000	1179/20/2500	1600/00/4000	8	16000
Standard_E_48bds_v5	48	384	1800	32	2250/00/2000	1200/00/4000	1200/00/4000	1600/00/4000	1600/00/4000	8	16000
Standard_E_64bds_v5	64	512	2400	32	3000/00/4000	1200/00/4000	1200/00/4000	1600/00/4000	1600/00/4000	8	20000

## Ebsv5 series

Ebsv5-series sizes run on the Intel® Xeon® Platinum 8272CL (Ice Lake). These VMs are ideal for memory-intensive enterprise applications and applications that benefit from high remote storage performance but with no local SSD storage. Ebsv5-series VMs feature Intel® Hyper-Threading Technology. Remote Data disk storage is billed separately from VMs.

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 1 and Generation 2
- [Accelerated Networking](#): Supported (required)
- [Ephemeral OS Disks](#): Not supported
- Nested virtualization: Supported

SIZE	VCPU	MEMORY: GiB	MAX DATA DISKS	MAX UNCAC HED PREMIUM V1 SSD AND STANDA RD SSD/HD D DISK THROU GHPUT: IOPS/M BPS	MAX UNCAC HED PREMIUM V1 SSD AND STANDA RD SSD/HD D DISK THROU GHPUT: IOPS/M BPS	MAX UNCAC HED ULTRA DISK AND PREMIUM V2 SSD DISK THROU GHPUT: IOPS/M BPS	MAX UNCAC HED ULTRA DISK AND PREMIUM V2 SSD DISK THROU GHPUT: IOPS/M BPS	MAX NICs	NETWORK BANDWIDTH
Standard_E2bs_v5	2	16	4	5500/156	10000/1200	7370/156	15000/1200	2	12500
Standard_E4bs_v5	4	32	8	11000/350	20000/1200	14740/350	30000/1200	2	12500
Standard_E8bs_v5	8	64	16	22000/625	40000/1200	29480/625	60000/1200	4	12500
Standard_E16bs_v5	16	128	32	44000/1250	64000/2000	58960/1250	96000/2000	8	12500
Standard_E32bs_v5	32	256	32	88000/2500	120000/4000	117920/2500	160000/4000	8	16000
Standard_E48bs_v5	48	384	32	120000/4000	120000/4000	160000/4000	160000/4000	8	16000
Standard_E64bs_v5	64	512	32	120000/4000	120000/4000	160000/4000	160000/4000	8	20000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

## Next steps

- Use the Azure [Pricing Calculator](#)

# Edv5 and Edsv5-series

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Edv5 and Edsv5-series Virtual Machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper threaded](#) configuration, providing a better value proposition for most general-purpose workloads. This new processor features an all core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#). Featuring up to 672 GiB of RAM, these virtual machines are ideal for memory-intensive enterprise applications, relational database servers, and in-memory analytics workloads. These VMs also feature fast and large local SSD storage (up to 3,900 GiB).

## Edv5-series

Edv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 104 vCPU and 672 GiB of RAM and fast, local SSD storage up to 3800 GiB. Edv5-series virtual machines are ideal for memory-intensive enterprise applications and applications that benefit from low latency, high-speed local storage.

Edv5-series virtual machines support Standard SSD and Standard HDD disk types. To use Premium SSD or Ultra Disk storage, select Edsv5-series virtual machines. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/Mbps *	MAX NICs	MAX NETWORK BANDWIDTH (Mbps)
Standard_E_2d_v5 <sup>1,2</sup>	2	16	75	4	9000/125	2	12500
Standard_E_4d_v5	4	32	150	8	19000/250	2	12500
Standard_E_8d_v5	8	64	300	16	38000/500	4	12500
Standard_E_16d_v5	16	128	600	32	75000/1000	8	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E 20d_v5	20	160	750	32	94000/1250	8	12500
Standard_E 32d_v5	32	256	1200	32	150000/2000	8	16000
Standard_E 48d_v5	48	384	1800	32	225000/3000	8	24000
Standard_E 64d_v5	64	512	2400	32	300000/4000	8	30000
Standard_E 96d_v5	96	672	3600	32	450000/4000	8	35000
Standard_E 104id_v5 <sup>3</sup>	104	672	3800	64	450000/4000	8	100000

\* These IOPs values can be guaranteed by using [Gen2 VMs](#)

<sup>1</sup> Accelerated networking is required and turned on by default on all Edv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> Instance is [isolated](#) to hardware dedicated to a single customer.

## Edsv5-series

Edsv5-series virtual machines run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor reaching an all core turbo clock speed of up to 3.5 GHz. These virtual machines offer up to 104 vCPU and 672 GiB of RAM and fast, local SSD storage up to 3800 GiB. Edsv5-series virtual machines are ideal for memory-intensive enterprise applications and applications that benefit from low latency, high-speed local storage.

Edsv5-series virtual machines support Standard SSD and Standard HDD disk types. You can attach Standard SSDs, Standard HDDs, and Premium SSDs disk storage to these VMs. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Required

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS*	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS <sup>5</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E2ds_v5 <sup>1,2</sup>	2	16	75	4	9000/125	3750/85	10000/1200	2	12500
Standard_E4ds_v5	4	32	150	8	19000/250	6400/145	20000/1200	2	12500
Standard_E8ds_v5	8	64	300	16	38000/500	12800/290	20000/1200	4	12500
Standard_E16ds_v5	16	128	600	32	75000/1000	25600/600	40000/1200	8	12500
Standard_E20ds_v5	20	160	750	32	94000/1250	32000/750	64000/1600	8	12500
Standard_E32ds_v5	32	256	1200	32	150000/2000	51200/865	80000/2000	8	16000
Standard_E48ds_v5	48	384	1800	32	225000/3000	76800/1315	80000/3000	8	24000
Standard_E64ds_v5	64	512	2400	32	375000/4000	80000/1735	80000/3000	8	30000
Standard_E96ds_v5 <sup>3</sup>	96	672	3600	32	450000/4000	80000/2600	80000/4000	8	35000
Standard_E104ids_v5 <sup>4,6</sup>	104	672	3800	64	450000/4000	120000/4000	120000/4000	8	100000

\* These IOPs values can be guaranteed by using [Gen2 VMs](#)

<sup>1</sup> Accelerated networking is required and turned on by default on all Edsv5 virtual machines.

<sup>2</sup> Accelerated networking can be applied to two NICs.

<sup>3</sup> [Constrained Core](#) sizes available.

<sup>4</sup> Instance is [isolated](#) to hardware dedicated to a single customer.

<sup>5</sup> Edsv5-series virtual machines can [burst](#) their disk performance and get up to their bursting max for up to 30

minutes at a time.

<sup>6</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_E104ids\_v5** results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 160000/4000
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 160000/4000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

# Easv5 and Eadsv5-series

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Easv5-series and Eadsv5-series utilize AMD's 3rd Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache, increasing customer options for running most memory optimized workloads. These virtual machines offer a combination of vCPUs and memory to meet the requirements associated with most memory-intensive enterprise applications, such as relational database servers and in-memory analytics workloads.

## Easv5-series

Easv5-series utilize AMD's 3rd Generation EPYC™ 7763v processors that can achieve a boosted maximum frequency of 3.5GHz. The Easv5-series sizes offer a combination of vCPU and memory that is ideal for memory-intensive enterprise applications. The new VMs with no local disk provide a better value proposition for workloads that do not require local temp disk.

### NOTE

For frequently asked questions, see [Azure VM sizes with no local temp disk](#).

Easv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Supported

[Memory Preserving Updates](#): Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS <sup>1</sup>	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E2as_v5	2	16	Remote Storage Only	4	3750/82	10000/600	2	12500
Standard_E4as_v5 <sup>2</sup>	4	32	Remote Storage Only	8	6400/144	20000/600	2	12500

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWOR K BANDWID TH (Mbps)
Standard_E8as_v5 <sup>2</sup>	8	64	Remote Storage Only	16	12800/200	20000/600	4	12500
Standard_E16as_v5 <sup>2</sup>	16	128	Remote Storage Only	32	25600/384	40000/800	8	12500
Standard_E20as_v5	20	160	Remote Storage Only	32	32000/480	64000/1000	8	12500
Standard_E32as_v5 <sup>2</sup>	32	256	Remote Storage Only	32	51200/768	80000/1600	8	16000
Standard_E48as_v5	48	384	Remote Storage Only	32	76800/1152	80000/2000	8	24000
Standard_E64as_v5 <sup>2</sup>	64	512	Remote Storage Only	32	80000/1200	80000/2000	8	32000
Standard_E96as_v5 <sup>2</sup>	96	672	Remote Storage Only	32	80000/1600	80000/2000	8	40000
Standard_E112ias_v5 <sup>3</sup>	112	672	Remote Storage Only	64	120000/2000	120000/2000	8	50000

<sup>1</sup> Easv5-series VMs can **burst** their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> **Constrained core sizes available**

<sup>3</sup> Attaching Ultra Disk or Premium v2 SSDs to Standard\_E112ias\_v5 results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 160000/2000
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 160000/2000

## Eadsv5-series

Eadsv5-series utilize AMD's 3rd Generation EPYC™ 7763v processors that can achieve a boosted maximum frequency of 3.5GHz. The Eadsv5-series sizes offer a combination of vCPU, memory and temporary storage that is ideal for memory-intensive enterprise applications. The new VMs have 50% larger local storage, as well as better local disk IOPS for both read and write compared to the [Eav4/Easv4](#) sizes with [Gen2](#) VMs.

Eadsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual

machines. [See pricing for disks](#).

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Supported

**Memory Preserving Updates:** Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS	MAX UNCAC HED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCAC HED DISK THROU GPUT: IOPS/M BPS <sup>1</sup>	MAX NICs	MAX NETWORK BANDWIDTH (MBPS)
Standard_E2ads_v5	2	16	75	4	9000 / 125	3750/82	10000/600	2	12500
Standard_E4ads_v5 <sup>2</sup>	4	32	150	8	19000 / 250	6400/144	20000/600	2	12500
Standard_E8ads_v5 <sup>2</sup>	8	64	300	16	38000 / 500	12800/200	20000/600	4	12500
Standard_E16ads_v5 <sup>2</sup>	16	128	600	32	75000 / 1000	25600/384	40000/800	8	12500
Standard_E20ads_v5	20	160	750	32	94000 / 1250	32000/480	64000/1000	8	12500
Standard_E32ads_v5 <sup>2</sup>	32	256	1200	32	150000 / 2000	51200/768	80000/1600	8	16000
Standard_E48ads_v5	48	384	1800	32	225000 / 3000	76800/1152	80000/2000	8	24000
Standard_E64ads_v5 <sup>2</sup>	64	512	2400	32	300000 / 4000	80000/1200	80000/2000	8	32000
Standard_E96ads_v5 <sup>2</sup>	96	672	3600	32	450000 / 4000	80000/1600	80000/2000	8	40000

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROU GPUT: IOPS/M BPS	MAX UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCACHED DISK THROU GPUT: IOPS/M BPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E112iads_v5 <sup>3</sup>	112	672	3800	64	450000 / 4000	120000 /2000	120000 /2000	8	50000

<sup>1</sup> Eadsv5-series VMs can [burst](#) their disk performance and get up to their bursting max for up to 30 minutes at a time.

<sup>2</sup> [Constrained core sizes available.](#)

<sup>3</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_E112iads\_v5** results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 160000/2000
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 160000/2000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)

- Previous generations

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# ECasv5 and ECadsv5-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

The ECasv5-series and ECadsv5-series are [confidential VMs](#) for use in Confidential Computing.

These confidential VMs use AMD's third-Generation EPYC™ 7763v processor in a multi-threaded configuration with up to 256 MB L3 cache. This processor can achieve a boosted maximum frequency of 3.5 GHz. Both series offer Secure Encrypted Virtualization-Secure Nested Paging (SEV-SNP). SEV-SNP provides hardware-isolated VMs that protect data from other VMs, the hypervisor, and host management code. Confidential VMs offer hardware-based VM memory encryption. These series also offer OS disk pre-encryption before VM provisioning with different key management solutions.

These VM series also offer a combination of vCPUs and memory to meet the requirements of most memory-intensive enterprise applications.

## ECasv5-series

ECasv5-series VMs offer a combination of vCPU and memory for memory-intensive enterprise applications.

These VMs with no local disk provide a better value proposition for workloads where you don't need a local temp disk. For more information, see the [FAQ for Azure VM sizes with no local temporary disk](#).

This series supports Standard SSD, Standard HDD, and Premium SSD disk types. Billing for disk storage and VMs is separate. To estimate your costs, use the [Pricing Calculator](#).

### NOTE

There are some [pricing differences based on your encryption settings](#) for confidential VMs.

### ECasv5-series feature support

*Supported* features in ECasv5-series VMs:

- [Premium Storage](#)
- [Premium Storage caching](#)
- [VM Generation 2](#)

*Unsupported* features in ECasv5-series VMs:

- [Live Migration](#)
- [Memory Preserving Updates](#)
- [Accelerated Networking](#)
- [Ephemeral OS Disks](#)
- [Nested Virtualization](#)

### ECasv5-series products

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_EC2as_v5	2	16	Remote Storage Only	4	3750/82	2
Standard_EC4as_v5	4	32	Remote Storage Only	8	6400/144	2
Standard_EC8as_v5	8	64	Remote Storage Only	16	12800/200	4
Standard_EC16as_v5	16	128	Remote Storage Only	32	25600/384	4
Standard_EC20as_v5	20	160	Remote Storage Only	32	32000/480	8
Standard_EC32as_v5	32	256	Remote Storage Only	32	51200/768	8
Standard_EC48as_v5	48	384	Remote Storage Only	32	76800/1152	8
Standard_EC64as_v5	64	512	Remote Storage Only	32	80000/1200	8
Standard_EC96as_v5	96	672	Remote Storage Only	32	80000/1600	8

## ECadsv5-series

ECadsv5-series VMs offer a combination of vCPU, memory, and temporary storage for memory-intensive enterprise applications. These VMs offer local storage.

This series supports Standard SSD, Standard HDD, and Premium SSD disk types. Billing for disk storage and VMs is separate. To estimate your costs, use the [Pricing Calculator](#).

### NOTE

There are some [pricing differences based on your encryption settings](#) for confidential VMs.

### ECadsv5-series feature support

*Supported* features in DCadsv5-series VMs:

- [Premium Storage](#)
- [Premium Storage caching](#)
- [VM Generation 2](#)

*Unsupported* features in DCadsv5-series VMs:

- [Live Migration](#)
- [Memory Preserving Updates](#)

- Accelerated Networking
- Ephemeral OS Disks

## ECadsv5-series products

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_E C2ads_v5	2	16	75	4	9000 / 125	3750/82	2
Standard_E C4ads_v5	4	32	150	8	19000 / 250	6400/144	2
Standard_E C8ads_v5	8	64	300	16	38000 / 500	12800/200	4
Standard_E C16ads_v5	16	128	600	32	75000 / 1000	25600/384	4
Standard_E C20ads_v5	20	160	750	32	94000 / 1250	32000/480	8
Standard_E C32ads_v5	32	256	1200	32	150000 / 2000	51200/768	8
Standard_E C48ads_v5	48	384	1800	32	225000 / 3000	76800/1152	8
Standard_E C64ads_v5	64	512	2400	32	300000 / 4000	80000/1200	8
Standard_E C96ads_v5	96	672	3600	32	450000 / 4000	80000/1600	8

### NOTE

To achieve these IOPs, use [Gen2 VMs](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk](#)

performance.

- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Next steps

[Confidential virtual machine options on AMD processors](#)

# Epsv5 and Epdsv5-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Epsv5-series and Epdsv5-series virtual machines are based on the Arm architecture, delivering outstanding price-performance for memory-intensive workloads. These virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer a range of vCPU sizes, up to 8 GiB of memory per vCPU, and are best suited for memory-intensive scale-out and enterprise workloads, such as relational database servers, large databases, data analytics engines, in-memory caches, and more.

## Epsv5-series

Epsv5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 32 vCPU and 208 GiB of RAM and are ideal for memory-intensive scale-out and most Enterprise workloads. Epsv5-series virtual machines support Standard SSD, Standard HDD, and Premium SSD disk types with no local-SSD support. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Not supported
- [Nested virtualization](#): Not supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX BURST UNCACH ED DISK THROUG HPUT: IOPS/MB PS	MAX NICS	MAX NETWORK BANDWID TH (MBPS)
Standard_E2ps_v5	2	16	Remote Storage Only	4	3750/85	10000/1200	2	12500
Standard_E4ps_v5	4	32	Remote Storage Only	8	6400/145	10000/1200	2	12500
Standard_E8ps_v5	8	64	Remote Storage Only	16	12800/290	20000/1200	4	12500

SIZE	vCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E16ps_v5	16	128	Remote Storage Only	32	25600/600	40000/1200	4	12500
Standard_E20ps_v5	20	160	Remote Storage Only	32	32000/750	64000/1600	8	12500
Standard_E32ps_v5	32	208	Remote Storage Only	32	51200/865	80000/2000	8	16000

#### NOTE

Accelerated networking is required and turned on by default on all Epsv5 machines.

## EpdsV5-series

EpdsV5-series virtual machines feature the Ampere® Altra® Arm-based processor operating at 3.0 GHz, which provides an entire physical core for each virtual machine vCPU. These virtual machines offer up to 32 vCPU, 208 GiB of RAM, and fast local SSD storage up to 1,200 GiB and are ideal for memory-intensive scale-out and most Enterprise workloads. EpdsV5-series virtual machines support Standard SSD, Standard HDD, and premium SSD disk types. You can also attach Ultra Disk storage based on its regional availability. Disk storage is billed separately from virtual machines. [See pricing for disks](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Live Migration](#): Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- [Nested virtualization](#): Not supported

SIZE	vCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT: IOPS/MBPS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX UNCACED DISK THROUGHPUT: IOPS/MBPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E2pds_v5	2	16	75	4	9375/125	3750/85	10000/1200	2	12500

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX TEMP THROU GPUT: IOPS/M BPS	MAX UNCAC HED DISK THROU GPUT: IOPS/M BPS	MAX BURST UNCAC HED DISK THROU GPUT: IOPS/M BPS	MAX NICS	MAX NETWORK BANDWIDTH (MBPS)
Standard_E4pds_v5	4	32	150	8	19000/250	6400/145	20000/1200	2	12500
Standard_E8pds_v5	8	64	300	16	38000/500	12800/290	20000/1200	4	12500
Standard_E16pds_v5	16	128	600	32	75000/1000	25600/600	40000/1200	4	12500
Standard_E20pds_v5	20	160	750	32	95000/1250	32000/750	64000/1600	8	12500
Standard_E32pds_v5	32	208	1200	32	150000/2000	51200/865	80000/2000	8	16000

#### NOTE

Accelerated networking is required and turned on by default on all Epsv5 machines.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see

Bandwidth/Throughput testing (NTTCP).

## Other sizes and information

- General purpose
- Memory optimized
- Storage optimized
- GPU optimized
- High performance compute
- Previous generations

Pricing Calculator: [Pricing Calculator](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# M-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The M-series offers a high vCPU count (up to 128 vCPUs) and a large amount of memory (up to 3.8 TiB). It's also ideal for extremely large databases or other applications that benefit from high vCPU counts and large amounts of memory. M-series sizes are supported both on the Intel® Xeon® CPU E7-8890 v3 @ 2.50GHz and on the Intel® Xeon® Platinum 8280M (Cascade Lake).

M-series VM's feature Intel® Hyper-Threading Technology.

**ACU:** 160-180

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Write Accelerator:** Supported

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**Nested Virtualization:** Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STORE GE THROU GPUT : IOPS/ MBPS (CACH E SIZE IN GIB)	BURST CACHE D AND TEMP STORE GE THROU GPUT : IOPS/ MBPS <sup>4</sup>	MAX UNCA CHED DISK THROU GPUT : IOPS/ MBPS	BURST UNCA CHED DISK THROU GPUT : IOPS/ MBPS <sup>4</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standar d_M8 ms <sup>3</sup>	8	218.75	256	8	10000/ 100 (793)	10000/ 250	5000/1 25	10000/ 250	4	2000
Standar d_M1 6ms <sup>3</sup>	16	437.5	512	16	20000/ 200 (1587)	20000/ 500	10000/ 250	20000/ 500	8	4000
Standar d_M3 2ts	32	192	1024	32	40000/ 400 (3174)	40000/ 1000	20000/ 500	40000/ 1000	8	8000
Standar d_M3 2ls	32	256	1024	32	40000/ 400 (3174)	40000/ 1000	20000/ 500	40000/ 1000	8	8000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHE D AND TEMP STOREAGE THROU GPUT : IOPS/ MBPS (CACH E SIZE IN GIB)	BURST CACHE D AND TEMP STOREAGE THROU GPUT : IOPS/ MBPS	MAX UNCA CHED DISK THROU GPUT : IOPS/ MBPS	BURST UNCA CHED DISK THROU GPUT : IOPS/ MBPS	MAX NICs	EXPECTED NETWORK BAND WIDTH (MBPS)
Standard_M3_2ms <sup>3</sup>	32	875	1024	32	40000/ 400 (3174)	40000/ 1000	20000/ 500	40000/ 1000	8	8000
Standard_M6_4s <sup>1</sup>	64	1024	2048	64	80000/ 800 (6348)	80000/ 2000	40000/ 1000	80000/ 2000	8	16000
Standard_M6_4ls <sup>1</sup>	64	512	2048	64	80000/ 800 (6348)	80000/ 2000	40000/ 1000	80000/ 2000	8	16000
Standard_M6_4ms <sub>1,3</sub>	64	1792	2048	64	80000/ 800 (6348)	80000/ 2000	40000/ 1000	80000/ 2000	8	16000
Standard_M1_28s <sup>1</sup>	128	2048	4096	64	16000 0/1600 (12696 )	25000 0/4000	80000/ 2000	80000/ 4000	8	30000
Standard_M1_28ms <sub>1,2,3</sub>	128	3892	4096	64	16000 0/1600 (12696 )	25000 0/4000	80000/ 2000	80000/ 4000	8	30000
Standard_M6_4 <sup>1</sup>	64	1024	7168	64	80000/ 800 (1228)	80000/ 2000	40000/ 1000	80000/ 2000	8	16000
Standard_M6_4m <sup>1</sup>	64	1792	7168	64	80000/ 800 (1228)	80000/ 2000	40000/ 1000	80000/ 2000	8	16000
Standard_M1_28 <sup>1</sup>	128	2048	14336	64	25000 0/1600 (2456)	25000 0/4000	80000/ 2000	80000/ 4000	8	32000
Standard_M1_28m <sup>1</sup>	128	3892	14336	64	25000 0/1600 (2456)	25000 0/4000	80000/ 2000	80000/ 4000	8	32000

<sup>1</sup> More than 64 vCPU's require one of these supported guest versions: Windows Server 2016, Ubuntu 16.04 LTS, SLES 12 SP2, and Red Hat Enterprise Linux, CentOS 7.3 or Oracle Linux 7.3 with LIS 4.2.1.

<sup>2</sup> Instance is isolated to hardware dedicated to a single customer.

<sup>3</sup> Constrained core sizes available.

<sup>4</sup> M-series VMs can **burst** their disk performance for up to 30 minutes at a time.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Msv2 and Mdsv2-series Medium Memory

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Msv2 and Mdsv2 Medium Memory VM Series features Intel® Xeon® Platinum 8280 (Cascade Lake) processor with an all core base frequency of 2.7 GHz and 4.0 GHz single core turbo frequency. With these VMs, customers achieve increased flexibility with local disk and diskless options. Customers also have access to a set of new isolated VM sizes with more CPU and memory that go up to 192 vCPU with 4 TiB of memory.

## NOTE

Msv2 and Mdsv2 Medium Memory VMs are generation 2 only. For more information on generation 2 virtual machines, see [Support for generation 2 VMs on Azure](#).

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 2

**Write Accelerator:** Supported

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported for Mdsv2

**Nested Virtualization:** Not Supported

## Msv2 Medium Memory Diskless

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS <sup>1</sup>	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_M32ms_v2	32	875	0	32	20000/500	40000/1000	8	8000
Standard_M64s_v2	64	1024	0	64	40000/1000	80000/2000	8	16000
Standard_M64ms_v2	64	1792	0	64	40000/1000	80000/2000	8	16000
Standard_M128s_v2	128	2048	0	64	80000/2000	80000/4000	8	30000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	BURST UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICS	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_M128ms_v2	128	3892	0	64	80000/2000	80000/4000	8	30000
Standard_M192is_v2 <sup>2</sup>	192	2048	0	64	80000/2000	80000/4000	8	30000
Standard_M192ims_v2	192	4096	0	64	80000/2000	80000/4000	8	30000

<sup>1</sup> Msv2 and Mdsv2 medium memory VMs can **burst** their disk performance for up to 30 minutes at a time.

<sup>2</sup> Attaching Ultra Disk or Premium v2 SSDs to **Standard\_M192is\_v2** results in higher IOPs and MBps than standard premium disks:

- Max uncached Ultra Disk and Premium v2 SSD throughput (IOPS/ MBps): 120000/2000
- Max burst uncached Ultra Disk and Premium v2 SSD disk throughput (IOPS/ MBps): 120000/4000

## Mdsv2 Medium Memory with Disk

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISK	MAX CACHE D AND TEMP STORAGE THROUGHPUT : IOPS / MBPS <sup>1</sup>	BURST CACHE D AND TEMP STORAGE THROUGHPUT : IOPS/ MBPS <sup>1</sup>	MAX UNCACHED DISK THROUGHPUT : IOPS/ MBPS	BURST UNCA CHED DISK THROUGHPUT : IOPS/ MBPS <sup>1</sup>	MAX NICS	EXPEC TED NETW ORK BAND WIDTH (MBPS)
Standard_M3_2dms_v2	32	875	1024	32	40000/400	40000/1000	20000/500	40000/1000	8	8000
Standard_M6_4ds_v2	64	1024	2048	64	80000/800	80000/2000	40000/1000	80000/2000	8	16000
Standard_M6_4dms_v2	64	1792	2048	64	80000/800	80000/2000	40000/1000	80000/2000	8	16000
Standard_M1_28ds_v2	128	2048	4096	64	16000 0/1600	25000 0/4000	80000/2000	80000/4000	8	30000

SIZE	VCPU	MEMORY: GIB	TEMP STORE GE (SSD) GIB	MAX DATA DISK	MAX CACHE D AND TEMP STORAGE THROU GHPUT : IOPS / MBPS	BURST CACHE D AND TEMP STORAGE THROU GHPUT :	MAX UNCA CHED DISK THROU GHPUT : IOPS/ MBPS	BURST UNCA CHED DISK THROU GHPUT : IOPS/ MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_M1_28dms_v2	128	3892	4096	64	16000 0/1600	25000 0/4000	80000/ 2000	80000/ 4000	8	30000
Standard_M1_92ids_v2	192	2048	4096	64	16000 0/1600	25000 0/4000	80000/ 2000	80000/ 4000	8	30000
Standard_M1_92idms_v2	192	4096	4096	64	16000 0/1600	25000 0/4000	80000/ 2000	80000/ 4000	8	30000

<sup>1</sup> Msv2 and Mdsv2 medium memory VMs can **burst** their disk performance for up to 30 minutes at a time.

## Size table definitions

- Storage capacity is shown in units of GiB or 1024^3 bytes. When you compare disks measured in GB (1000^3 bytes) to disks measured in GiB (1024^3) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps = 10^6 bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)

- GPU optimized
- High performance compute
- Previous generations

Pricing Calculator: [Pricing Calculator](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Mv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Mv2-series features high throughput, low latency platform running on a hyper-threaded Intel® Xeon® Platinum 8180M 2.5GHz (Skylake) processor with an all core base frequency of 2.5 GHz and a max turbo frequency of 3.8 GHz. All Mv2-series virtual machine sizes can use both standard and premium persistent disks. Mv2-series instances are memory optimized VM sizes providing unparalleled computational performance to support large in-memory databases and workloads, with a high memory-to-CPU ratio that is ideal for relational database servers, large caches, and in-memory analytics.

Mv2-series VM's feature Intel® Hyper-Threading Technology

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 2

[Write Accelerator](#): Supported

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS / MBPS (CACHE SIZE IN GIB)	MAX UNCACHED DISK THROUGHPUT: IOPS / MBPS	MAX NICs	EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_M208ms_v2 <sup>1</sup>	208	5700	4096	64	80000 / 800 (7040)	40000 / 1000	8	16000
Standard_M208s_v2 <sup>1</sup>	208	2850	4096	64	80000 / 800 (7040)	40000 / 1000	8	16000
Standard_M416ms_v2 <sup>1,2</sup>	416	11400	8192	64	250000 / 1600 (14080)	80000 / 2000	8	32000
Standard_M416s_v2 <sup>1,2</sup>	416	5700	8192	64	250000 / 1600 (14080)	80000 / 2000	8	32000

<sup>1</sup> Mv2-series VMs are generation 2 only and support a subset of generation 2 supported Images. Please see below for the complete list of supported images for Mv2-series. If you're using Linux, see [Support for](#)

[generation 2 VMs on Azure](#) for instructions on how to find and select an image. If you're using Windows, see [Support for generation 2 VMs on Azure](#) for instructions on how to find and select an image.

- Windows Server 2019 or later
- SUSE Linux Enterprise Server 12 SP4 and later or SUSE Linux Enterprise Server 15 SP1 and later
- Red Hat Enterprise Linux 7.6 or later, and 8.1 or later
- Oracle Enterprise Linux 7.7 or later, and 8.1 or later
- Ubuntu 18.04 with the 5.4.0-azure kernel or later

<sup>2</sup> [Constrained core sizes available](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types : [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Constrained vCPU capable VM sizes

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## TIP

Try the [Virtual Machine selector tool](#) to find other sizes that best fit your workload.

Some database workloads like SQL Server require high memory, storage, and I/O bandwidth, but not a high core count. Many database workloads are not CPU-intensive. Azure offers certain VM sizes where you can lower the VM vCPU count to reduce the cost of software licensing, while maintaining the same memory, storage, and I/O bandwidth.

The available vCPU count can be reduced to one half or one quarter of the original VM specification. These new VM sizes have a suffix that specifies the number of available vCPUs to make them easier for you to identify. There are no additional cores available that can be used by the VM.

For example, the current VM size Standard\_E32s\_v5 comes with 32 vCPUs, 256 GiB RAM, 32 disks, and 80,000 IOPs or 2 GB/s of I/O bandwidth. The new VM sizes Standard\_E32-16s\_v5 and Standard\_E32-8s\_v5 comes with 16 and 8 active vCPUs respectively, while maintaining the rest of the specs of the Standard\_E32s\_v5 for memory, storage, and I/O bandwidth.

The licensing fees charged for SQL Server are based on the avaialble vCPU count. Third party products should count the available vCPU which represents the max to be used and licensed. This results in a 50% to 75% increase in the ratio of the VM specs to available (billable) vCPUs. These new VM sizes allow customer workloads to use the same memory, storage, and I/O bandwidth while optimizing their software licensing cost. At this time, the compute cost, which includes OS licensing, remains the same one as the original size. For more information, see [Azure VM sizes for more cost-effective database workloads](#).

NAME	VCPUs	SPECS
Standard_M8-2ms	2	Same as M8ms
Standard_M8-4ms	4	Same as M8ms
Standard_M16-4ms	4	Same as M16ms
Standard_M16-8ms	8	Same as M16ms
Standard_M32-8ms	8	Same as M32ms
Standard_M32-16ms	16	Same as M32ms
Standard_M64-32ms	32	Same as M64ms
Standard_M64-16ms	16	Same as M64ms
Standard_M128-64ms	64	Same as M128ms

NAME	VCPUs	SPECS
Standard_M128-32ms	32	Same as M128ms
Standard_E4-2s_v3	2	Same as E4s_v3
Standard_E8-4s_v3	4	Same as E8s_v3
Standard_E8-2s_v3	2	Same as E8s_v3
Standard_E16-8s_v3	8	Same as E16s_v3
Standard_E16-4s_v3	4	Same as E16s_v3
Standard_E32-16s_v3	16	Same as E32s_v3
Standard_E32-8s_v3	8	Same as E32s_v3
Standard_E64-32s_v3	32	Same as E64s_v3
Standard_E64-16s_v3	16	Same as E64s_v3
Standard_E4-2s_v4	2	Same as E4s_v4
Standard_E8-4s_v4	4	Same as E8s_v4
Standard_E8-2s_v4	2	Same as E8s_v4
Standard_E16-8s_v4	8	Same as E16s_v4
Standard_E16-4s_v4	4	Same as E16s_v4
Standard_E32-16s_v4	16	Same as E32s_v4
Standard_E32-8s_v4	8	Same as E32s_v4
Standard_E64-32s_v4	32	Same as E64s_v4
Standard_E64-16s_v4	16	Same as E64s_v4
Standard_E4-2ds_v4	2	Same as E4ds_v4
Standard_E8-4ds_v4	4	Same as E8ds_v4
Standard_E8-2ds_v4	2	Same as E8ds_v4
Standard_E16-8ds_v4	8	Same as E16ds_v4
Standard_E16-4ds_v4	4	Same as E16ds_v4
Standard_E32-16ds_v4	16	Same as E32ds_v4

NAME	VCPUs	SPECS
Standard_E32-8ds_v4	8	Same as E32ds_v4
Standard_E64-32ds_v4	32	Same as E64ds_v4
Standard_E64-16ds_v4	16	Same as E64ds_v4
Standard_E4-2s_v5	2	Same as E4s_v5
Standard_E8-4s_v5	4	Same as E8s_v5
Standard_E8-2s_v5	2	Same as E8s_v5
Standard_E16-8s_v5	8	Same as E16s_v5
Standard_E16-4s_v5	4	Same as E16s_v5
Standard_E32-16s_v5	16	Same as E32s_v5
Standard_E32-8s_v5	8	Same as E32s_v5
Standard_E64-32s_v5	32	Same as E64s_v5
Standard_E64-16s_v5	16	Same as E64s_v5
Standard_E96-48s_v5	48	Same as E96s_v5
Standard_E96-24s_v5	24	Same as E96s_v5
Standard_E4-2ds_v5	2	Same as E4ds_v5
Standard_E8-4ds_v5	4	Same as E8ds_v5
Standard_E8-2ds_v5	2	Same as E8ds_v5
Standard_E16-8ds_v5	8	Same as E16ds_v5
Standard_E16-4ds_v5	4	Same as E16ds_v5
Standard_E32-16ds_v5	16	Same as E32ds_v5
Standard_E32-8ds_v5	8	Same as E32ds_v5
Standard_E64-32ds_v5	32	Same as E64ds_v5
Standard_E64-16ds_v5	16	Same as E64ds_v5
Standard_E96-48ds_v5	48	Same as E96ds_v5
Standard_E96-24ds_v5	24	Same as E96ds_v5

NAME	VCPUs	SPECS
Standard_E4-2as_v4	2	Same as E4as_v4
Standard_E8-4as_v4	4	Same as E8as_v4
Standard_E8-2as_v4	2	Same as E8as_v4
Standard_E16-8as_v4	8	Same as E16as_v4
Standard_E16-4as_v4	4	Same as E16as_v4
Standard_E32-16as_v4	16	Same as E32as_v4
Standard_E32-8as_v4	8	Same as E32as_v4
Standard_E64-32as_v4	32	Same as E64as_v4
Standard_E64-16as_v4	16	Same as E64as_v4
Standard_E96-48as_v4	48	Same as E96as_v4
Standard_E96-24as_v4	24	Same as E96as_v4
Standard_E4-2ads_v5	2	Same as E4ads_v5
Standard_E8-4ads_v5	4	Same as E8ads_v5
Standard_E8-2ads_v5	2	Same as E8ads_v5
Standard_E16-8ads_v5	8	Same as E16ads_v5
Standard_E16-4ads_v5	4	Same as E16ads_v5
Standard_E32-16ads_v5	16	Same as E32ads_v5
Standard_E32-8ads_v5	8	Same as E32ads_v5
Standard_E64-32ads_v5	32	Same as E64ads_v5
Standard_E64-16ads_v5	16	Same as E64ads_v5
Standard_E96-48ads_v5	48	Same as E96ads_v5
Standard_E96-24ads_v5	24	Same as E96ads_v5
Standard_E4-2as_v5	2	Same as E4as_v5
Standard_E8-4as_v5	4	Same as E8as_v5
Standard_E8-2as_v5	2	Same as E8as_v5

NAME	VCPUs	SPECS
Standard_E16-8as_v5	8	Same as E16as_v5
Standard_E16-4as_v5	4	Same as E16as_v5
Standard_E32-16as_v5	16	Same as E32as_v5
Standard_E32-8as_v5	8	Same as E32as_v5
Standard_E64-32as_v5	32	Same as E64as_v5
Standard_E64-16as_v5	16	Same as E64as_v5
Standard_E96-48as_v5	48	Same as E96as_v5
Standard_E96-24as_v5	24	Same as E96as_v5
Standard_GS4-8	8	Same as GS4
Standard_GS4-4	4	Same as GS4
Standard_GS5-16	16	Same as GS5
Standard_GS5-8	8	Same as GS5
Standard_DS11-1_v2	1	Same as DS11_v2
Standard_DS12-2_v2	2	Same as DS12_v2
Standard_DS12-1_v2	1	Same as DS12_v2
Standard_DS13-4_v2	4	Same as DS13_v2
Standard_DS13-2_v2	2	Same as DS13_v2
Standard_DS14-8_v2	8	Same as DS14_v2
Standard_DS14-4_v2	4	Same as DS14_v2
Standard_M416-208s_v2	208	Same as M416s_v2
Standard_M416-208ms_v2	208	Same as M416ms_v2

## Other sizes

- [Compute optimized](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU](#)
- [High performance compute](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Storage optimized virtual machine sizes

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Storage optimized virtual machine (VM) sizes offer high disk throughput and IO, and are ideal for Big Data, SQL, NoSQL databases, data warehousing, and large transactional databases. Examples include Cassandra, MongoDB, Cloudera, and Redis. This article provides information about the number of vCPUs, data disks, NICs, local storage throughput, and network bandwidth for each optimized size.

## TIP

Try the [virtual machines selector tool](#) to find other sizes that best fit your workload.

The Lsv3, Lasv3, and Lsv2-series feature high-throughput, low latency, directly mapped local NVMe storage. These VM series come in sizes from 8 to 80 vCPU. There are 8 GiB of memory per vCPU, and one 1.92TB NVMe SSD device per 8 vCPUs, with up to 19.2TB (10x1.92TB) available on the largest VM sizes.

- The [Lsv3-series](#) runs on the third Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper-threaded configuration](#). This new processor features an all-core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#).
- The [Lasv3-series](#) runs on the AMD 3rd Generation EPYC™ 7763v processor. This series runs in a multi-threaded configuration with up to 256 MB L3 cache, which can achieve a boosted maximum frequency of 3.5 GHz.
- The [Lsv2-series](#) runs on the [AMD EPYC™ 7551 processor](#) with an all-core boost of 2.55 GHz and a max boost of 3.0 GHz.

## Other sizes

- [General purpose](#)
- [Compute optimized](#)
- [Memory optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

## Next steps

- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.
- Learn how to optimize performance on the Lsv2-series [Windows VMs](#) and [Linux VMs](#).
- For more information on how Azure names its VMs, see [Azure virtual machine sizes naming conventions](#).

# Lsv2-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale sets

The Lsv2-series features high throughput, low latency, directly mapped local NVMe storage running on the [AMD EPYC™ 7551 processor](#) with an all core boost of 2.55GHz and a max boost of 3.0GHz. The Lsv2-series VMs come in sizes from 8 to 80 vCPU in a simultaneous multi-threading configuration. There is 8 GiB of memory per vCPU, and one 1.92TB NVMe SSD M.2 device per 8 vCPUs, with up to 19.2TB (10x1.92TB) available on the L80s v2.

## NOTE

The Lsv2-series VMs are optimized to use the local disk on the node attached directly to the VM rather than using durable data disks. This allows for greater IOPs / throughput for your workloads. The Lsv2 and Ls-series do not support the creation of a local cache to increase the IOPs achievable by durable data disks.

The high throughput and IOPs of the local disk makes the Lsv2-series VMs ideal for NoSQL stores such as Apache Cassandra and MongoDB which replicate data across multiple VMs to achieve persistence in the event of the failure of a single VM.

To learn more, see Optimize performance on the Lsv2-series virtual machines for [Windows](#) or [Linux](#).

[ACU: 150-175](#)

[Premium Storage](#): Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

Bursting: Supported

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY (GiB)	TEMP DISK <sup>1</sup> (GiB)	NVME DISKS <sup>2</sup>	NVME DISK THROU GHPUT <sup>3</sup> (READ IOPS/ MBPS)	UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS) <sup>4</sup>	MAX BURST UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS) <sup>5</sup>	MAX DATA DISKS	MAX NICs	EXPEC TED NETW ORK BAND WIDTH (MBPS)
Standard_L8s_v2	8	64	80	1x1.92 TB	40000 0/2000	8000/160	8000/1280	16	2	3200
Standard_L16s_v2	16	128	160	2x1.92 TB	80000 0/4000	16000/320	16000/1280	32	4	6400

SIZE	VCPUs	MEMORY (GiB)	TEMP DISK (GiB)	NVME DISKS	NVME DISK THROUGHPUT (READ IOPS/Mbps)	UNCACHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAXBURST UNCACHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_L32s_v2	32	256	320	4x1.92 TB	1.5M/8000	32000/640	32000/1280	32	8	12800
Standard_L48s_v2	48	384	480	6x1.92 TB	2.2M/14000	48000/960	48000/2000	32	8	16000+
Standard_L64s_v2	64	512	640	8x1.92 TB	2.9M/16000	64000/1280	64000/2000	32	8	16000+
Standard_L80s_v2 <sup>6</sup>	80	640	800	10x1.92TB	3.8M/20000	80000/1400	80000/2000	32	8	16000+

<sup>1</sup> Lsv2-series VMs have a standard SCSI based temp resource disk for OS paging/swap file use (D: on Windows, /dev/sdb on Linux). This disk provides 80 GiB of storage, 4,000 IOPS, and 80 MBps transfer rate for every 8 vCPUs (e.g. Standard\_L80s\_v2 provides 800 GiB at 40,000 IOPS and 800 MBPS). This ensures the NVMe drives can be fully dedicated to application use. This disk is Ephemeral, and all data will be lost on stop/deallocate.

<sup>2</sup> Local NVMe disks are ephemeral, data will be lost on these disks if you stop/deallocate your VM. Local NVMe disks aren't encrypted by [Azure Storage encryption](#), even if you enable [encryption at host](#).

<sup>3</sup> Hyper-V NVMe Direct technology provides unthrottled access to local NVMe drives mapped securely into the guest VM space. Achieving maximum performance requires using either the latest WS2019 build or Ubuntu 18.04 or 16.04 from the Azure Marketplace. Write performance varies based on IO size, drive load, and capacity utilization.

<sup>4</sup> Lsv2-series VMs do not provide host cache for data disk as it does not benefit the Lsv2 workloads.

<sup>5</sup> Lsv2-series VMs can [burst](#) their disk performance for up to 30 minutes at a time.

<sup>6</sup> VMs with more than 64 vCPUs require one of these supported guest operating systems:

- Windows Server 2016 or later
- Ubuntu 16.04 LTS or later, with Azure tuned kernel (4.15 kernel or later)
- SLES 12 SP2 or later
- RHEL or CentOS version 6.7 through 6.10, with Microsoft-provided LIS package 4.3.1 (or later) installed
- RHEL or CentOS version 7.3, with Microsoft-provided LIS package 4.2.1 (or later) installed
- RHEL or CentOS version 7.6 or later
- Oracle Linux with UEF4 or later
- Debian 9 with the backports kernel, Debian 10 or later
- CoreOS with a 4.14 kernel or later

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When comparing disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- If you want to get the best performance for your VMs, you should limit the number of data disks to 2 disks per vCPU.
- **Expected network bandwidth** is the maximum aggregated [bandwidth allocated per VM type](#) across all NICs, for all destinations. Upper limits are not guaranteed, but are intended to provide guidance for selecting the right VM type for the intended application. Actual network performance will depend on a variety of factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimizing network throughput for Windows and Linux](#). To achieve the expected network performance on Linux or Windows, it may be necessary to select a specific version or optimize your VM. For more information, see [How to reliably test for virtual machine throughput](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Lsv3-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Lsv3-series of Azure Virtual Machines (Azure VMs) features high-throughput, low latency, directly mapped local NVMe storage. These VMs run on the 3rd Generation Intel® Xeon® Platinum 8370C (Ice Lake) processor in a [hyper-threaded configuration](#). This new processor features an all-core turbo clock speed of 3.5 GHz with [Intel® Turbo Boost Technology](#), [Intel® Advanced-Vector Extensions 512 \(Intel® AVX-512\)](#) and [Intel® Deep Learning Boost](#).

The Lsv3-series VMs are available in sizes from 8 to 80 vCPUs. There are 8 GiB of memory allocated per vCPU, and one 1.92TB NVMe SSD device allocated per 8 vCPUs, with up to 19.2TB (10x1.92TB) available on the L80s\_v3 size.

## NOTE

The Lsv3-series VMs are optimized to use the local disk on the node attached directly to the VM rather than using [durable data disks](#). This method allows for greater IOPS and throughput for your workloads. The Lsv3, Lasv3, Lsv2, and Ls-series VMs don't support the creation of a host cache to increase the IOPS achievable by durable data disks.

The high throughput and IOPS of the local disk makes the Lsv3-series VMs ideal for NoSQL stores such as Apache Cassandra and MongoDB. These stores replicate data across multiple VMs to achieve persistence in the event of the failure of a single VM.

To learn more, see how to optimize performance on the Lsv3-series [Windows-based VMs](#) or [Linux-based VMs](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Not Supported
- [Live Migration](#): Not Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 1 and 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- [Nested Virtualization](#): Supported

SIZE	VCPU	MEMO RY (GiB)	TEMP DISK (GiB)	NVME DISKS	NVME DISK THROU GHPUT (READ IOPS/ MBPS)	UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS)	MAX BURST UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS)	MAX DATA DISKS	MAX NICs	EXPEC TED NETW ORK BAND WIDTH (Mbps)
Standar d_L8s _v3	8	64	80	1x1.92 TB	40000 0/2000	12800/ 290	20000/ 1200	16	4	12500

SIZE	VCPUs	MEMORY (GiB)	TEMP DISK (GiB)	NVME DISKS	NVME DISK THROUGHPUT (READ IOPS/Mbps)	UNCA CHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAX BURST UNCA CHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_L16s_v3	16	128	160	2x1.92 TB	80000/0/4000	25600/600	40000/1600	32	8	12500
Standard_L32s_v3	32	256	320	4x1.92 TB	1.5M/8000	51200/865	80000/2000	32	8	16000
Standard_L48s_v3	48	384	480	6x1.92 TB	2.2M/14000	76800/1315	80000/3000	32	8	24000
Standard_L64s_v3	64	512	640	8x1.92 TB	2.9M/16000	80000/1735	80000/3000	32	8	30000
Standard_L80s_v3	80	640	800	10x1.92TB	3.8M/20000	80000/2160	80000/3000	32	8	32000

- Temp disk:** Lsv3-series VMs have a standard SCSI-based temp resource disk for use by the OS paging or swap file (`D:` on Windows, `/dev/sdb` on Linux). This disk provides 80 GiB of storage, 4,000 IOPS, and 80 MBps transfer rate for every 8 vCPUs. For example, Standard\_L80s\_v3 provides 800 GiB at 40000 IOPS and 800 MBPS. This configuration ensures the NVMe drives can be fully dedicated to application use. This disk is ephemeral, and all data is lost on stop or deallocation.
- NVMe Disks:** NVMe disk throughput can go higher than the specified numbers. However, higher performance isn't guaranteed. Local NVMe disks are ephemeral. Data is lost on these disks if you stop or deallocate your VM. Local NVMe disks aren't encrypted by [Azure Storage encryption](#), even if you enable [encryption at host](#).
- NVMe Disk throughput:** Hyper-V NVMe Direct technology provides unthrottled access to local NVMe drives mapped securely into the guest VM space. Lsv3 NVMe disk throughput can go higher than the specified numbers, but higher performance isn't guaranteed. To achieve maximum performance, see how to optimize performance on the Lsv3-series [Windows-based VMs](#) or [Linux-based VMs](#). Read/write performance varies based on IO size, drive load, and capacity utilization.
- Max burst uncached data disk throughput:** Lsv3-series VMs can [burst their disk performance](#) for up to 30 minutes at a time.

#### NOTE

Lsv3-series VMs don't provide host cache for data disk as it doesn't benefit the Lsv3 workloads.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may

appear smaller. For example, 1023 GiB = 1098.4 GB.

- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Lasv3-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The Lasv3-series of Azure Virtual Machines (Azure VMs) features high-throughput, low latency, directly mapped local NVMe storage. These VMs run on an AMD 3rd Generation EPYC™ 7763v processor in a multi-threaded configuration with an L3 cache of up to 256 MB that can achieve a boosted maximum frequency of 3.5 GHz. The Lasv3-series VMs are available in sizes from 8 to 80 vCPUs in a simultaneous multi-threading configuration. There are 8 GiB of memory per vCPU, and one 1.92 TB NVMe SSD device per 8 vCPUs, with up to 19.2 TB (10x1.92TB) available on the L80as\_v3 size.

## NOTE

The Lasv3-series VMs are optimized to use the local disk on the node attached directly to the VM rather than using [durable data disks](#). This method allows for greater IOPS and throughput for your workloads. The Lsv3, Lasv3, Lsv2, and Ls-series don't support the creation of a local cache to increase the IOPS achievable by durable data disks.

The high throughput and IOPS of the local disk makes the Lasv3-series VMs ideal for NoSQL stores such as Apache Cassandra and MongoDB. These stores replicate data across multiple VMs to achieve persistence in the event of the failure of a single VM.

To learn more, see how optimize performance on Lasv3-series [Windows-based VMs](#) or [Linux-based VMs](#).

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Not Supported
- [Live Migration](#): Not Supported
- [Memory Preserving Updates](#): Supported
- [VM Generation Support](#): Generation 1 and 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- [Nested Virtualization](#): Supported

SIZE	VCPU	MEMORY (GiB)	TEMP DISK (GiB)	NVME DISKS	NVME DISK THROU GHPUT (READ IOPS/ MBPS)	UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS)	MAX BURST UNCA CHED DATA DISK THROU GHPUT (IOPS/ MBPS)	MAX DATA DISKS	MAX NICs	EXPEC TED NETW ORK BAND WIDTH (MBPS)
Standar d_L8a s_v3	8	64	80	1x1.92 TB	40000 0/2000	12800/ 200	20000/ 1280	16	4	12500
Standar d_L16 as_v3	16	128	160	2x1.92 TB	80000 0/4000	25600/ 384	40000/ 1280	32	8	12500

SIZE	VCPUs	MEMORY (GiB)	TEMP DISK (GiB)	NVME DISKS	NVME DISK THROUGHPUT (READ IOPS/Mbps)	UNCA CHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAX BURST UNCA CHED DATA DISK THROUGHPUT (IOPS/Mbps)	MAX DATA DISKS	MAX NICs	EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_L32as_v3	32	256	320	4x1.92 TB	1.5M/8000	51200/768	80000/1600	32	8	16000
Standard_L48as_v3	48	384	480	6x1.92 TB	2.2M/14000	76800/1152	80000/2000	32	8	24000
Standard_L64as_v3	64	512	640	8x1.92 TB	2.9M/16000	80000/1280	80000/2000	32	8	32000
Standard_L80as_v3	80	640	800	10x1.92TB	3.8M/20000	80000/1400	80000/2000	32	8	32000

- Temp disk:** Lasv3-series VMs have a standard SCSI-based temp resource disk for use by the OS paging or swap file (`D:` on Windows, `/dev/sdb` on Linux). This disk provides 80 GiB of storage, 4000 IOPS, and 80 MBps transfer rate for every 8 vCPUs. For example, Standard\_L80as\_v3 provides 800 GiB at 40000 IOPS and 800 MBPS. This configuration ensures that the NVMe drives can be fully dedicated to application use. This disk is ephemeral, and all data is lost on stop or deallocation.
- NVMe Disks:** NVMe disk throughput can go higher than the specified numbers. However, higher performance isn't guaranteed. Local NVMe disks are ephemeral. Data is lost on these disks if you stop or deallocate your VM. Local NVMe disks aren't encrypted by [Azure Storage encryption](#), even if you enable [encryption at host](#).
- NVMe Disk throughput:** Hyper-V NVMe Direct technology provides unthrottled access to local NVMe drives mapped securely into the guest VM space. Lasv3 NVMe disk throughput can go higher than the specified numbers, but higher performance isn't guaranteed. To achieve maximum performance, see how to optimize performance on Lasv3-series [Windows-based VMs](#) or [Linux-based VMs](#). Read/write performance varies based on IO size, drive load, and capacity utilization.
- Max burst uncached data disk throughput:** Lasv3-series VMs can [burst their disk performance](#) for up to 30 minutes at a time.

#### NOTE

Lasv3-series VMs don't provide a host cache for the data disk because this configuration doesn't benefit the Lasv3 workloads.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =

$10^6$  bytes/sec.

- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

More information on Disks Types: [Disk Types](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Optimize performance on Lsv3, Lasv3, and Lsv2-series Linux VMs

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Uniform scale sets

Lsv3, Lasv3, and Lsv2-series Azure Virtual Machines (Azure VMs) support various workloads that need high I/O and throughput on local storage across a wide range of applications and industries. The L-series is ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases, including Cassandra, MongoDB, Cloudera, and Redis.

Several builds are available Azure Marketplace due to work with partners in Linux. These builds are optimized for Lsv3, Lasv3, and Lsv2-series performance. Available builds include the following and later versions of:

- Ubuntu 16.04
- RHEL 8.0 and clones, including CentOS, Rocky Linux, and Alma Linux
- Debian 9
- SUSE Linux 15
- Oracle Linux 8.0

This article provides tips and suggestions to ensure your workloads and applications achieve the maximum performance designed into the VMs.

## AMD EPYC™ chipset architecture

Lasv3 and Lsv2-series VMs use AMD EPYC™ server processors based on the Zen micro-architecture. AMD developed Infinity Fabric (IF) for EPYC™ as scalable interconnect for its NUMA model that can be used for on-die, on-package, and multi-package communications. Compared with QPI (Quick-Path Interconnect) and UPI (Ultra-Path Interconnect) used on Intel modern monolithic-die processors, AMD's many-NUMA small-die architecture can bring both performance benefits and challenges. The actual effects of memory bandwidth and latency constraints might vary depending on the type of workloads running.

## Tips to maximize performance

- If you're uploading a custom Linux GuestOS for your workload, Accelerated Networking is turned off by default. If you intend to enable Accelerated Networking, enable it at the time of VM creation for best performance.
- To gain max performance, run multiple jobs with deep queue depth per device.
- Avoid mixing NVMe admin commands (for example, NVMe SMART info query, etc.) with NVMe I/O commands during active workloads. Lsv3, Lasv3, and Lsv2 NVMe devices are backed by Hyper-V NVMe Direct technology, which switches into "slow mode" whenever any NVMe admin commands are pending. Lsv3, Lasv3, and Lsv2 users might see a dramatic performance drop in NVMe I/O performance if that happens.
- Lsv2 users aren't recommended to rely on device NUMA information (all 0) reported from within the VM for data drives to decide the NUMA affinity for their apps. The recommended way for better performance is to spread workloads across CPUs if possible.
- The maximum supported queue depth per I/O queue pair for Lsv3, Lasv3, and Lsv2 VM NVMe device is 1024. Lsv3, Lasv3, and Lsv2 users are recommended to limit their (synthetic) benchmarking workloads to queue depth 1024 or lower to avoid triggering queue full conditions, which can reduce performance.

- The best performance is obtained when I/O is done directly to each of the raw NVMe devices with no partitioning, no file systems, no RAID config, etc. Before starting a testing session, ensure the configuration is in a known fresh/clean state by running `blkdiscard` on each of the NVMe devices.

## Utilizing local NVMe storage

Local storage on the 1.92 TB NVMe disk on all Lsv3, Lasv3, and Lsv2 VMs is ephemeral. During a successful standard reboot of the VM, the data on the local NVMe disk persists. The data doesn't persist on the NVMe if the VM is redeployed, de-allocated, or deleted. Data doesn't persist if another issue causes the VM, or the hardware it's running on, to become unhealthy. When scenario happens, any data on the old host is securely erased.

There are also cases when the VM needs to be moved to a different host machine, for example, during a planned maintenance operation. Planned maintenance operations and some hardware failures can be anticipated with [Scheduled Events](#). Use Scheduled Events to stay updated on any predicted maintenance and recovery operations.

In the case that a planned maintenance event requires the VM to be recreated on a new host with empty local disks, the data needs to be resynchronized (again, with any data on the old host being securely erased). This scenario occurs because Lsv3, Lasv3, and Lsv2-series VMs don't currently support live migration on the local NVMe disk.

There are two modes for planned maintenance.

### **Standard VM customer-controlled maintenance**

- The VM is moved to an updated host during a 30-day window.
- Lsv3, Lasv3, and Lsv2 local storage data could be lost, so backing-up data prior to the event is recommended.

### **Automatic maintenance**

- Occurs if the customer doesn't execute customer-controlled maintenance, or because of emergency procedures, such as a security zero-day event.
- Intended to preserve customer data, but there's a small risk of a VM freeze or reboot.
- Lsv3, Lasv3, and Lsv2 local storage data could be lost, so backing-up data prior to the event is recommended.

For any upcoming service events, use the controlled maintenance process to select a time most convenient to you for the update. Prior to the event, back up your data in premium storage. After the maintenance event completes, you can return your data to the refreshed Lsv3, Lasv3, and Lsv2 VMs local NVMe storage.

Scenarios that maintain data on local NVMe disks include:

- The VM is running and healthy.
- The VM is rebooted in place (by you or Azure).
- The VM is paused (stopped without de-allocation).
- Most the planned maintenance servicing operations.

Scenarios that securely erase data to protect the customer include:

- The VM is redeployed, stopped (de-allocated), or deleted (by you).
- The VM becomes unhealthy and has to service heal to another node due to a hardware issue.
- A few of the planned maintenance servicing operations that require the VM to be reallocated to another host for servicing.

## Frequently asked questions

The following are frequently asked questions about these series.

### **How do I start deploying L-series VMs?**

Much like any other VM, use the [Portal](#), [Azure CLI](#), or [PowerShell](#) to create a VM.

### **Does a single NVMe disk failure cause all VMs on the host to fail?**

If a disk failure is detected on the hardware node, the hardware is in a failed state. When this problem occurs, all VMs on the node are automatically de-allocated and moved to a healthy node. For Lsv3, Lasv3, and Lsv2-series VMs, this problem means that the customer's data on the failing node is also securely erased. The customer needs to recreate the data on the new node.

### **Do I need to change the blk\_mq settings?**

RHEL/CentOS 7.x automatically uses blk-mq for the NVMe devices. No configuration changes or settings are necessary.

## **Next steps**

See specifications for all [VMs optimized for storage performance](#) on Azure

# Optimize performance on Lsv3, Lasv3, and Lsv2-series Windows VMs

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Uniform scale sets

Lsv3, Lasv3, and Lsv2-series Azure Virtual Machines (Azure VMs) support various workloads that need high I/O and throughput on local storage across a wide range of applications and industries. The L-series is ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases, including Cassandra, MongoDB, Cloudera, and Redis.

Lsv3, Lasv3, and Lsv2-series VMs are designed to work with the needs of Windows and Linux operating systems for better performance with hardware and the software.

Software and hardware tuning resulted in the optimized version of [Windows Server 2019 Datacenter](#), released to the Azure Marketplace (and later versions), which support maximum performance on the NVMe devices in L-series VMs.

This article provides tips and suggestions to ensure your workloads and applications achieve the maximum performance designed into the VMs.

## AMD EPYC™ chipset architecture

Lasv3 and Lsv2-series VMs use AMD EPYC™ server processors based on the Zen micro-architecture. AMD developed Infinity Fabric (IF) for EPYC™ as a scalable interconnect for its NUMA model that can be used for on-die, on-package, and multi-package communications. Compared with QPI (Quick-Path Interconnect) and UPI (Ultra-Path Interconnect), used on Intel modern monolithic-die processors, AMD's many-NUMA small-die architecture can bring both performance benefits and challenges. The actual effects of memory bandwidth and latency constraints can vary depending on the type of workloads.

## Tips for maximizing performance

- To gain max performance, run multiple jobs with deep queue depth per device.
- Avoid mixing NVMe admin commands (for example, NVMe SMART info query) with NVMe I/O commands during active workloads. Lsv3, Lasv3, and Lsv2 NVMe devices are backed by Hyper-V NVMe Direct technology, which switches into "slow mode" whenever any NVMe admin commands are pending. Lsv3, Lasv3, and Lsv2 users might see a dramatic performance drop in NVMe I/O performance if that scenario happens.
- It's not recommended for Lsv2 users to rely on device NUMA information (all 0) reported from within the VM for data drives to decide the NUMA affinity for their apps. For better performance, it's recommended to spread workloads across CPUs if possible.
- The maximum supported queue depth per I/O queue pair for Lsv3, Lasv3, and Lsv2 VM NVMe device is 1024. Lsv3, Lasv3, and Lsv2 users are recommended to limit their (synthetic) benchmarking workloads to queue depth 1024 or lower to avoid triggering queue full conditions, which can reduce performance.
- The best performance is obtained when I/O is done directly to each of the raw NVMe devices with no partitioning, no file systems, no RAID config, etc.

# Utilizing local NVMe storage

Local storage on the 1.92 TB NVMe disk on all Lsv3, Lasv3, and Lsv2 VMs is ephemeral. During a successful standard reboot of the VM, the data on the local NVMe disk persists. The data doesn't persist on the NVMe if the VM is redeployed, deallocated, or deleted. Data doesn't persist if another issue causes the VM, or the hardware on which the VM is running, to become unhealthy. When this scenario happens, any data on the old host is securely erased.

There are also cases when the VM needs to be moved to a different host machine; for example, during a planned maintenance operation. Planned maintenance operations and some hardware failures can be anticipated with [Scheduled Events](#). Use Scheduled Events to stay updated on any predicted maintenance and recovery operations.

In the case that a planned maintenance event requires the VM to be recreated on a new host with empty local disks, the data needs to be resynchronized (again, with any data on the old host being securely erased). This scenario occurs because Lsv3, Lasv3, and Lsv2-series VMs don't currently support live migration on the local NVMe disk.

There are two modes for planned maintenance: [standard VM customer-controlled maintenance](#) and [automatic maintenance](#).

For any upcoming service events, use the controlled maintenance process to select a time most convenient to you for the update. Prior to the event, back up your data in premium storage. After the maintenance event completes, return your data to the refreshed Lsv2 VMs local NVMe storage.

Scenarios that maintain data on local NVMe disks include when:

- The VM is running and healthy.
- The VM is rebooted in place by you or by Azure.
- The VM is paused (stopped without deallocation).
- Most planned maintenance servicing operations.

Scenarios that securely erase data to protect the customer include when:

- The VM is redeployed, stopped (deallocated), or deleted by you.
- The VM becomes unhealthy and has to service heal to another node due to a hardware issue.
- A few the planned maintenance servicing operations that require the VM to be reallocated to another host for servicing.

## **Standard VM customer-controlled maintenance**

In standard VM customer-controlled maintenance, the VM is moved to an updated host during a 30-day window.

Lsv3, Lasv3, and Lsv2 local storage data might be lost, so backing-up data prior to the event is recommended.

## **Automatic maintenance**

Automatic maintenance occurs if the customer doesn't execute customer-controlled maintenance. Automatic maintenance can also occur because of emergency procedures, such as a security zero-day event.

This type of maintenance is intended to preserve customer data, but there's a small risk of a VM freeze or reboot.

Lsv3, Lasv3, and Lsv2 local storage data might be lost, so backing-up data prior to the event is recommended.

## Frequently asked questions

The following are frequently asked questions about these series.

## **How do I start deploying L-series VMs?**

Much like any other VM, create a VM using the [Azure portal](#), [through the Azure Command-Line Interface \(Azure CLI\)](#), or [through PowerShell](#).

## **Does a single NVMe disk failure cause all VMs on the host to fail?**

If a disk failure is detected on the hardware node, the hardware is in a failed state. When this problem occurs, all VMs on the node are automatically deallocated and moved to a healthy node. For Lsv3, Lasv3, and Lsv2-series VMs, this scenario means that the customer's data on the failing node is also securely erased. The customer needs to recreate the data on the new node.

## **Do I need to make polling adjustments in Windows Server 2012 or Windows Server 2016?**

NVMe polling is only available on Windows Server 2019 and later versions on Azure.

## **Can I switch back to a traditional interrupt service routine (ISR) model?**

Lasv3, and Lsv2-series VMs are optimized for NVMe polling. Updates are continuously provided to improve polling performance.

## **Can I adjust the polling settings in Windows Server 2019 or later versions?**

The polling settings aren't user adjustable.

## **Next steps**

See specifications for all [VMs optimized for storage performance](#) on Azure.

# GPU optimized virtual machine sizes

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

GPU optimized VM sizes are specialized virtual machines available with single, multiple, or fractional GPUs. These sizes are designed for compute-intensive, graphics-intensive, and visualization workloads. This article provides information about the number and type of GPUs, vCPUs, data disks, and NICs. Storage throughput and network bandwidth are also included for each size in this grouping.

- The [NCv3-series](#) and [NC T4\\_v3-series](#) sizes are optimized for compute-intensive GPU-accelerated applications. Some examples are CUDA and OpenCL-based applications and simulations, AI, and Deep Learning. The NC T4 v3-series is focused on inference workloads featuring NVIDIA's Tesla T4 GPU and AMD EPYC2 Rome processor. The NCv3-series is focused on high-performance computing and AI workloads featuring NVIDIA's Tesla V100 GPU.
- The [ND A100 v4-series](#) size is focused on scale-up and scale-out deep learning training and accelerated HPC applications. The ND A100 v4-series uses 8 NVIDIA A100 TensorCore GPUs, each available with a 200 Gigabit Mellanox InfiniBand HDR connection and 40 GB of GPU memory.
- [NV-series](#) and [NVv3-series](#) sizes are optimized and designed for remote visualization, streaming, gaming, encoding, and VDI scenarios using frameworks such as OpenGL and DirectX. These VMs are backed by the NVIDIA Tesla M60 GPU.
- [NVv4-series](#) VM sizes optimized and designed for VDI and remote visualization. With partitioned GPUs, NVv4 offers the right size for workloads requiring smaller GPU resources. These VMs are backed by the AMD Radeon Instinct MI25 GPU. NVv4 VMs currently support only Windows guest operating system.
- [NDm A100 v4-series](#) virtual machine is a new flagship addition to the Azure GPU family, designed for high-end Deep Learning training and tightly-coupled scale-up and scale-out HPC workloads. The NDm A100 v4 series starts with a single virtual machine (VM) and eight NVIDIA Ampere A100 80GB Tensor Core GPUs.

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA or AMD GPU drivers must be installed.

- For VMs backed by NVIDIA GPUs, the [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

Alternatively, you may install NVIDIA GPU drivers manually. See [Install NVIDIA GPU drivers on N-series VMs running Windows](#) or [Install NVIDIA GPU drivers on N-series VMs running Linux](#) for supported operating systems, drivers, installation, and verification steps.

- For VMs backed by AMD GPUs, the [AMD GPU driver extension](#) installs appropriate AMD drivers. Install or

manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

Alternatively, you may install AMD GPU drivers manually. See [Install AMD GPU drivers on N-series VMs running Windows](#) for supported operating systems, drivers, installation, and verification steps.

## Deployment considerations

- For availability of N-series VMs, see [Products available by region](#).
- N-series VMs can only be deployed in the Resource Manager deployment model.
- N-series VMs differ in the type of Azure Storage they support for their disks. NC and NV VMs only support VM disks that are backed by Standard Disk Storage (HDD). All other GPU VMs support VM disks that are backed by Standard Disk Storage and Premium Disk Storage (SSD).
- If you want to deploy more than a few N-series VMs, consider a pay-as-you-go subscription or other purchase options. If you're using an [Azure free account](#), you can use only a limited number of Azure compute cores.
- You might need to increase the cores quota (per region) in your Azure subscription, and increase the separate quota for NC, NCv2, NCv3, ND, NDv2, NV, or NVv2 cores. To request a quota increase, [open an online customer support request](#) at no charge. Default limits may vary depending on your subscription category.

## Other sizes

- [General purpose](#)
- [Compute optimized](#)
- [High performance compute](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [Previous generations](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NC-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

NC-series VMs are powered by the [NVIDIA Tesla K80](#) card and the Intel Xeon E5-2690 v3 (Haswell) processor. Users can crunch through data faster by leveraging CUDA for energy exploration applications, crash simulations, ray traced rendering, deep learning, and more. The NC24r configuration provides a low latency, high-throughput network interface optimized for tightly coupled parallel computing workloads.

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Not Supported

Nvidia NVLink Interconnect: Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICs
Standard_N_C6	6	56	340	1	12	24	1
Standard_N_C12	12	112	680	2	24	48	2
Standard_N_C24	24	224	1440	4	48	64	4
Standard_N_C24r*	24	224	1440	4	48	64	4

1 GPU = one-half K80 card.

\*RDMA capable

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NCv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

NCv2-series VMs are powered by NVIDIA Tesla P100 GPUs. These GPUs can provide more than 2x the computational performance of the NC-series. Customers can take advantage of these updated GPUs for traditional HPC workloads such as reservoir modeling, DNA sequencing, protein analysis, Monte Carlo simulations, and others. In addition to the GPUs, the NCv2-series VMs are also powered by Intel Xeon E5-2690 v4 (Broadwell) CPUs.

The NC24rs v2 configuration provides a low latency, high-throughput network interface optimized for tightly coupled parallel computing workloads.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Supported

Nvidia NVLink Interconnect: Not Supported

[Nested Virtualization](#): Not Supported

## IMPORTANT

For this VM series, the vCPU (core) quota in your subscription is initially set to 0 in each region. [Request a vCPU quota increase](#) for this series in an [available region](#).

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_NC6s_v2	6	112	736	1	16	12	20000/200	4
Standard_NC12s_v2	12	224	1474	2	32	24	40000/400	8
Standard_NC24s_v2	24	448	2948	4	64	32	80000/800	8
Standard_NC24rs_v2*	24	448	2948	4	64	32	80000/800	8

1 GPU = one P100 card.

\*RDMA capable

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NCv3-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

NCv3-series VMs are powered by NVIDIA Tesla V100 GPUs. These GPUs can provide 1.5x the computational performance of the NCv2-series. Customers can take advantage of these updated GPUs for traditional HPC workloads such as reservoir modeling, DNA sequencing, protein analysis, Monte Carlo simulations, and others. The NC24rs v3 configuration provides a low latency, high-throughput network interface optimized for tightly coupled parallel computing workloads. In addition to the GPUs, the NCv3-series VMs are also powered by Intel Xeon E5-2690 v4 (Broadwell) CPUs.

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Not Supported

**Ephemeral OS Disks:** Supported

Nvidia NVLink Interconnect: Not Supported

**Nested Virtualization:** Not Supported

## IMPORTANT

For this VM series, the vCPU (core) quota in your subscription is initially set to 0 in each region. [Request a vCPU quota increase](#) for this series in an [available region](#). These SKUs aren't available to trial or Visual Studio Subscriber Azure subscriptions. Your subscription level might not support selecting or deploying these SKUs.

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUROUGHPUT: IOPS/MBPS	MAX NICs
Standard_NC6s_v3	6	112	736	1	16	12	20000/200	4
Standard_NC12s_v3	12	224	1474	2	32	24	40000/400	8
Standard_NC24s_v3	24	448	2948	4	64	32	80000/800	8
Standard_NC24rs_v3*	24	448	2948	4	64	32	80000/800	8

1 GPU = one V100 card.

\*RDMA capable

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NCasT4\_v3-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NCasT4\_v3-series virtual machines are powered by [Nvidia Tesla T4](#) GPUs and AMD EPYC 7V12(Rome) CPUs. The VMs feature up to 4 NVIDIA T4 GPUs with 16 GB of memory each, up to 64 non-multithreaded AMD EPYC 7V12 (Rome) processor cores(base frequency of 2.45 GHz, all-cores peak frequency of 3.1 GHz and single-core peak frequency of 3.3 GHz) and 440 GiB of system memory. These virtual machines are ideal for deploying AI services- such as real-time inferencing of user-generated requests, or for interactive graphics and visualization workloads using NVIDIA's GRID driver and virtual GPU technology. Standard GPU compute workloads based around CUDA, TensorRT, Caffe, ONNX and other frameworks, or GPU-accelerated graphical applications based on OpenGL and DirectX can be deployed economically, with close proximity to users, on the NCasT4\_v3 series.

[ACU](#): 230-260

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage, and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

Nvidia NVLink Interconnect: Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	GPU	GPU MEMORY: GiB	MAX DATA DISKS	MAX NICs / EXPECTED NETWORK BANDWIDT H (MBPS)
Standard_N C4as_T4_v3	4	28	180	1	16	8	2 / 8000
Standard_N C8as_T4_v3	8	56	360	1	16	16	4 / 8000
Standard_N C16as_T4_v 3	16	110	360	1	16	32	8 / 8000
Standard_N C64as_T4_v 3	64	440	2880	4	64	32	8 / 32000

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure NCasT4\_v3-series VMs running Windows or Linux, Nvidia GPU drivers must be installed.

To install Nvidia GPU drivers manually, see [N-series GPU driver setup for Windows](#) for supported operating systems, drivers, installation, and verification steps.

The Azure Nvidia GPU driver extension will deploy CUDA drivers on the NCasT4\_v3-series VMs. For graphics and visualization workloads manually install the GRID drivers supported by Azure.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NC A100 v4-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NC A100 v4 series virtual machine (VM) is a new addition to the Azure GPU family. You can use this series for real-world Azure Applied AI training and batch inference workloads.

The NC A100 v4 series is powered by NVIDIA A100 PCIe GPU and 3rd-generation AMD EPYC™ 7V13 (Milan) processors. The VMs feature up to 4 NVIDIA A100 PCIe GPUs with 80GB memory each, up to 96 non-multithreaded AMD EPYC Milan processor cores and 880 GiB of system memory. These VMs are ideal for real-world Applied AI workloads, such as:

- GPU-accelerated analytics and databases
- Batch inferencing with heavy pre- and post-processing
- Autonomy model training
- Oil and gas reservoir simulation
- Machine learning (ML) development
- Video processing
- AI/ML web services

## Supported features

To get started with NC A100 v4 VMs, refer to [HPC Workload Configuration and Optimization](#) for steps including driver and network configuration.

Due to increased GPU memory I/O footprint, the NC A100 v4 requires the use of [Generation 2 VMs](#) and marketplace images. While the [Azure HPC images](#) are strongly recommended, Azure HPC Ubuntu 18.04, 20.04 and Azure HPC CentOS 7.9, CentOS 8.4, RHEL 7.9, RHEL 8.5, Windows Service 2019, and Windows Service 2022 images are supported.

Note: The Ubuntu-HPC 18.04-ncv4 image is only valid during preview and deprecated on 7/29/2022. All changes have been merged into standard Ubuntu-HPC 18.04 image. Please follow instruction [Azure HPC images](#) for configuration.

- [Premium Storage](#): Supported
- [Premium Storage caching](#): Supported
- [Ultra Disks](#): Not Supported
- [Live Migration](#): Not Supported
- [Memory Preserving Updates](#): Not Supported
- [VM Generation Support](#): Generation 2
- [Accelerated Networking](#): Supported
- [Ephemeral OS Disks](#): Supported
- InfiniBand: Not Supported
- Nvidia NVLink Interconnect: Supported
- [Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (WITH NVME) : GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS / MBPS	MAX NICs/NETWORK BANDWIDTH (MBPS)
Standard_NC24ads_A100_v4	24	220	1123	1	80	12	30000/1000	2/20,000
Standard_NC48ads_A100_v4	48	440	2246	2	160	24	60000/2000	4/40,000
Standard_NC96ads_A100_v4	96	880	4492	4	320	32	120000/4000	8/80,000

1 GPU = one A100 card

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

You can [use the pricing calculator](#) to estimate your Azure VMs costs.

For more information on disk types, see [What disk types are available in Azure?](#)

## Next step

- [Compare compute performance across Azure SKUs with Azure compute units \(ACU\)](#)

# ND A100 v4-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The ND A100 v4 series virtual machine is a new flagship addition to the Azure GPU family, designed for high-end Deep Learning training and tightly-coupled scale-up and scale-out HPC workloads.

The ND A100 v4 series starts with a single virtual machine (VM) and eight NVIDIA Ampere A100 40GB Tensor Core GPUs. ND A100 v4-based deployments can scale up to thousands of GPUs with an 1.6 Tb/s of interconnect bandwidth per VM. Each GPU within the VM is provided with its own dedicated, topology-agnostic 200 Gb/s NVIDIA Mellanox HDR InfiniBand connection. These connections are automatically configured between VMs occupying the same virtual machine scale set, and support GPUDirect RDMA.

Each GPU features NVLINK 3.0 connectivity for communication within the VM, and the instance is also backed by 96 physical 2nd-generation AMD Epyc™ 7V12 (Rome) CPU cores.

These instances provide excellent performance for many AI, ML, and analytics tools that support GPU acceleration 'out-of-the-box,' such as TensorFlow, Pytorch, Caffe, RAPIDS, and other frameworks. Additionally, the scale-out InfiniBand interconnect is supported by a large set of existing AI and HPC tools built on NVIDIA's NCCL2 communication libraries for seamless clustering of GPUs.

## IMPORTANT

To get started with ND A100 v4 VMs, refer to [HPC Workload Configuration and Optimization](#) for steps including driver and network configuration. Due to increased GPU memory I/O footprint, the ND A100 v4 requires the use of [Generation 2 VMs](#) and marketplace images. The [Azure HPC images](#) are strongly recommended. Azure HPC Ubuntu 18.04, 20.04 and Azure HPC CentOS 7.9 images are supported.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage, and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

InfiniBand: Supported, GPUDirect RDMA, 8 x 200 Gigabit HDR

Nvidia NVLink Interconnect: Supported

[Nested Virtualization](#): Not Supported

The ND A100 v4 series supports the following kernel versions:

CentOS 7.9 HPC: 3.10.0-1160.24.1.el7.x86\_64

Ubuntu 18.04: 5.4.0-1043-azure

Ubuntu 20.04: 5.4.0-1046-azure

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD): GiB	GPU	GPU MEMORY: GiB	MAX DATA DISKS	MAX UNCACHED DISK THROU GPUT: IOPS / MBPS	MAX NETWORK BANDWI DTH	MAX NICs
Standard_ND96asr_v4	96	900	6000	8 A100 40 GB GPUs (NVLink 3.0)	40	32	80,000 / 800	24,000 Mbps	8

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.



# NDm A100 v4-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NDM A100 v4 series virtual machine is a new flagship addition to the Azure GPU family, designed for high-end Deep Learning training and tightly-coupled scale-up and scale-out HPC workloads.

The NDM A100 v4 series starts with a single virtual machine (VM) and eight NVIDIA Ampere A100 80GB Tensor Core GPUs. NDM A100 v4-based deployments can scale up to thousands of GPUs with an 1.6 Tb/s of interconnect bandwidth per VM. Each GPU within the VM is provided with its own dedicated, topology-agnostic 200 Gb/s NVIDIA Mellanox HDR InfiniBand connection. These connections are automatically configured between VMs occupying the same virtual machine scale set, and support GPUDirect RDMA.

Each GPU features NVLINK 3.0 connectivity for communication within the VM, and the instance is also backed by 96 physical 2nd-generation AMD Epyc™ 7V12 (Rome) CPU cores.

These instances provide excellent performance for many AI, ML, and analytics tools that support GPU acceleration 'out-of-the-box,' such as TensorFlow, Pytorch, Caffe, RAPIDS, and other frameworks. Additionally, the scale-out InfiniBand interconnect is supported by a large set of existing AI and HPC tools built on NVIDIA's NCCL2 communication libraries for seamless clustering of GPUs.

## IMPORTANT

To get started with NDM A100 v4 VMs, refer to [HPC Workload Configuration and Optimization](#) for steps including driver and network configuration. Due to increased GPU memory I/O footprint, the NDM A100 v4 requires the use of [Generation 2 VMs](#) and marketplace images. The [Azure HPC images](#) are strongly recommended. Azure HPC Ubuntu 18.04, 20.04 and Azure HPC CentOS 7.9 images are supported.

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage, and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

InfiniBand: Supported, GPUDirect RDMA, 8 x 200 Gigabit HDR

Nvidia NVLink Interconnect: Supported

**Nested Virtualization:** Not Supported

The NDM A100 v4 series supports the following kernel versions:

CentOS 7.9 HPC: 3.10.0-1160.24.1.el7.x86\_64

Ubuntu 18.04: 5.4.0-1043-azure

Ubuntu 20.04: 5.4.0-1046-azure

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD): GiB	GPU	GPU MEMORY: GiB	MAX DATA DISKS	MAX UNCACHED DISK THROU GPUT: IOPS / MBPS	MAX NETWORK BANDWI DTH	MAX NICs
Standard_ND96amsr_A100_v4	96	1900	6400	8 A100 80 GB GPUs (NVLink 3.0)	80	32	80,000 / 800	24,000 Mbps	8

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.



# ND-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The ND-series virtual machines are a new addition to the GPU family designed for AI, and Deep Learning workloads. They offer excellent performance for training and inference. ND instances are powered by [NVIDIA Tesla P40](#) GPUs and Intel Xeon E5-2690 v4 (Broadwell) CPUs. These instances provide excellent performance for single-precision floating point operations, for AI workloads utilizing Microsoft Cognitive Toolkit, TensorFlow, Caffe, and other frameworks. The ND-series also offers a much larger GPU memory size (24 GB), enabling to fit much larger neural net models. Like the NC-series, the ND-series offers a configuration with a secondary low-latency, high-throughput network through RDMA, and InfiniBand connectivity so you can run large-scale training jobs spanning many GPUs.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Supported

Nvidia NVLink Interconnect: Not Supported

[Nested Virtualization](#): Not Supported

## IMPORTANT

For this VM series, the vCPU (core) quota per region in your subscription is initially set to 0. [Request a vCPU quota increase](#) for this series in an [available region](#).

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_ND6s	6	112	736	1	24	12	20000/200	4
Standard_ND12s	12	224	1474	2	48	24	40000/400	8
Standard_ND24s	24	448	2948	4	96	32	80000/800	8
Standard_ND24rs*	24	448	2948	4	96	32	80000/800	8

1 GPU = one P40 card.

\*RDMA capable

# Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure

SKUs.

# Updated NDv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NDv2-series virtual machine is a new addition to the GPU family designed for the needs of the most demanding GPU-accelerated AI, machine learning, simulation, and HPC workloads.

NDv2 is powered by 8 NVIDIA Tesla V100 NVLINK-connected GPUs, each with 32 GB of GPU memory. Each NDv2 VM also has 40 non-HyperThreaded Intel Xeon Platinum 8168 (Skylake) cores and 672 GiB of system memory.

NDv2 instances provide excellent performance for HPC and AI workloads utilizing CUDA GPU-optimized computation kernels, and the many AI, ML, and analytics tools that support GPU acceleration 'out-of-box,' such as TensorFlow, Pytorch, Caffe, RAPIDS, and other frameworks.

Critically, the NDv2 is built for both computationally intense scale-up (harnessing 8 GPUs per VM) and scale-out (harnessing multiple VMs working together) workloads. The NDv2 series now supports 100-Gigabit InfiniBand EDR backend networking, similar to that available on the HB series of HPC VM, to allow high-performance clustering for parallel scenarios including distributed training for AI and ML. This backend network supports all major InfiniBand protocols, including those employed by NVIDIA's NCCL2 libraries, allowing for seamless clustering of GPUs.

## IMPORTANT

When [enabling InfiniBand](#) on the ND40rs\_v2 VM, please use the 4.7-1.0.0.1 Mellanox OFED driver.

Due to increased GPU memory, the new ND40rs\_v2 VM requires the use of [Generation 2 VMs](#) and marketplace images.

Please note: The ND40s\_v2 featuring 16 GB of per-GPU memory is no longer available for preview and has been superceded by the updated ND40rs\_v2.

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 2

**Accelerated Networking:** Supported

**Ephemeral OS Disks:** Supported

**InfiniBand:** Supported

**Nvidia NVLink Interconnect:** Supported

**Nested Virtualization:** Not Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD): GiB	GPU	GPU MEMORY: GiB	MAX DATA DISKS	MAX UNCACHED DISK THROU GPUT: IOPS / MBPS	MAX NETWORK BANDWI DTH	MAX NICs
Standard_ND40rs_v2	40	672	2948	8 V100 32 GB (NVLink)	32	32	80000 / 800	24000 Mbps	8

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Linux](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NV-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## IMPORTANT

NV and NV\_Promo series Azure virtual machines (VMs) will be retired on August 31st, 2023. For more information, see the [NV and NV\\_Promo retirement information](#). For how to migrate your workloads to other VM sizes, see the [NV and NV\\_Promo series migration guide](#).

This retirement announcement doesn't apply to NVv3 and NVv4 series VMs.

The NV-series virtual machines are powered by [NVIDIA Tesla M60](#) GPUs and NVIDIA GRID technology for desktop accelerated applications and virtual desktops where customers are able to visualize their data or simulations. Users are able to visualize their graphics intensive workflows on the NV instances to get superior graphics capability and additionally run single precision workloads such as encoding and rendering. NV-series VMs are also powered by Intel Xeon E5-2690 v3 (Haswell) CPUs.

Each GPU in NV instances comes with a GRID license. This license gives you the flexibility to use an NV instance as a virtual workstation for a single user, or 25 concurrent users can connect to the VM for a virtual application scenario.

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Not Supported

[Ephemeral OS Disks](#): Not Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICs	VIRTUAL WORKSTATIONS	VIRTUAL APPLICATIONS
Standard_NV6	6	56	340	1	8	24	1	1	25
Standard_NV12	12	112	680	2	16	48	2	2	50
Standard_NV24	24	224	1440	4	32	64	4	4	100

1 GPU = one-half M60 card.

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NVv3-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NVv3-series virtual machines are powered by [NVIDIA Tesla M60](#) GPUs and NVIDIA GRID technology with Intel E5-2690 v4 (Broadwell) CPUs and Intel Hyper-Threading Technology. These virtual machines are targeted for GPU accelerated graphics applications and virtual desktops where customers want to visualize their data, simulate results to view, work on CAD, or render and stream content. Additionally, these virtual machines can run single precision workloads such as encoding and rendering. NVv3 virtual machines support Premium Storage and come with twice the system memory (RAM) when compared with its predecessor NV-series.

Each GPU in NVv3 instances comes with a GRID license. This license gives you the flexibility to use an NV instance as a virtual workstation for a single user, or 25 concurrent users can connect to the VM for a virtual application scenario.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD): GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCA CHED DISK THROU GHPUT : IOPS/ MBPS	MAX NICS / EXPEC TED NETW ORK BAND WIDTH (MBPS)	VIRTU AL WORKS TATIO NS	VIRTU AL APPLIC ATION S
Standard_NV 12s_v3	12	112	320	1	8	12	20000/ 200	4 / 6000	1	25
Standard_NV 24s_v3	24	224	640	2	16	24	40000/ 400	8 / 12000	2	50
Standard_NV 48s_v3	48	448	1280	4	32	32	80000/ 800	8 / 24000	4	100

<sup>1</sup> 1 GPU = one-half M60 card.

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure N-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NVv4-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NVv4-series virtual machines are powered by [AMD Radeon Instinct MI25](#) GPUs and AMD EPYC 7V12(Rome) CPUs with a base frequency of 2.45GHz, all-cores peak frequency of 3.1GHz and single-core peak frequency of 3.3GHz. With NVv4-series Azure is introducing virtual machines with partial GPUs. Pick the right sized virtual machine for GPU accelerated graphics applications and virtual desktops starting at 1/8th of a GPU with 2 GiB frame buffer to a full GPU with 16 GiB frame buffer. NVv4 virtual machines currently support only Windows guest operating system.

[ACU](#): 230-260

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICs / EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_N_V4as_v4	4	14	88	1/8	2	4	2 / 1000
Standard_N_V8as_v4	8	28	176	1/4	4	8	4 / 2000
Standard_N_V16as_v4	16	56	352	1/2	8	16	8 / 4000
Standard_N_V32as_v4	32	112	704	1	16	32	8 / 8000

<sup>1</sup> NVv4-series VMs feature AMD Simultaneous multithreading Technology

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure NVv4-series VMs running Windows, AMD GPU drivers must be installed.

To install AMD GPU drivers manually, see [N-series AMD GPU driver setup for Windows](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NVadsA10 v5-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NVadsA10v5-series virtual machines are powered by [NVIDIA A10](#) GPUs and AMD EPYC 74F3V(Milan) CPUs with a base frequency of 3.2 GHz, all-cores peak frequency of 4.0 GHz. With NVadsA10v5-series Azure is introducing virtual machines with partial NVIDIA GPUs. Pick the right sized virtual machine for GPU accelerated graphics applications and virtual desktops starting at 1/6th of a GPU with 4-GiB frame buffer to a full A10 GPU with 24-GiB frame buffer.

Each virtual machine instance in NVadsA10v5-series comes with a GRID license. This license gives you the flexibility to use an NV instance as a virtual workstation for a single user, or 25 concurrent users can connect to the VM for a virtual application scenario.

[ACU](#): Not Available

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Ultra Disks](#): Supported ([Learn more](#) about availability, usage and performance)

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	GPU PARTITION	GPU MEMORY: GiB	MAX DATA DISKS	MAX NICs / EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_NV6ads_A10_v5	6	55	180	1/6	4	4	2 / 5000
Standard_NV12ads_A10_v5	12	110	360	1/3	8	4	2 / 10000
Standard_NV18ads_A10_v5	18	220	720	1/2	12	8	4 / 20000
Standard_NV36ads_A10_v5	36	440	720	1	24	16	4 / 40000

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU PARTITION	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICs / EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_N_V36adms_A10_v5	36	880	720	1	24	32	8 / 80000
Standard_N_V72ads_A10_v5	72	880	1400	2	48	32	8 / 80000

<sup>1</sup> NVadsA10v5-series VMs feature AMD Simultaneous multithreading Technology

<sup>2</sup> The actual GPU VRAM reported in the operating system will be little less due to Error Correcting Code (ECC) support.

## Supported operating systems and drivers

To take advantage of the GPU capabilities of Azure NVadsA10v5-series VMs, NVIDIA GPU drivers must be installed.

The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps. For general information about VM extensions, see [Azure virtual machine extensions and features](#).

If you choose to install NVIDIA GPU drivers manually, see [N-series GPU driver setup for Windows](#) or [N-series GPU driver setup for Linux](#) for supported operating systems, drivers, installation, and verification steps.

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see

Bandwidth/Throughput testing (NTTCP).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Install NVIDIA GPU drivers on N-series VMs running Linux

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

To take advantage of the GPU capabilities of Azure N-series VMs backed by NVIDIA GPUs, you must install NVIDIA GPU drivers. The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as the Azure CLI or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported distributions and deployment steps.

If you choose to install NVIDIA GPU drivers manually, this article provides supported distributions, drivers, and installation and verification steps. Manual driver setup information is also available for [Windows VMs](#).

For N-series VM specs, storage capacities, and disk details, see [GPU Linux VM sizes](#).

## Supported distributions and drivers

### NVIDIA CUDA drivers

For the latest CUDA drivers and supported operating systems, visit the [NVIDIA](#) website. Ensure that you install or upgrade to the latest supported CUDA drivers for your distribution.

#### NOTE

The latest supported CUDA drivers for NC-series VMs is currently 470.82.01. Later driver versions are not supported on the K80 cards in NC.

#### TIP

As an alternative to manual CUDA driver installation on a Linux VM, you can deploy an Azure [Data Science Virtual Machine](#) image. The DSVM editions for Ubuntu 16.04 LTS or CentOS 7.4 pre-install NVIDIA CUDA drivers, the CUDA Deep Neural Network Library, and other tools.

### NVIDIA GRID drivers

Microsoft redistributes NVIDIA GRID driver installers for NV and NVv3-series VMs used as virtual workstations or for virtual applications. Install only these GRID drivers on Azure NV VMs, only on the operating systems listed in the following table. These drivers include licensing for GRID Virtual GPU Software in Azure. You do not need to set up a NVIDIA vGPU software license server.

The GRID drivers redistributed by Azure do not work on most non-NV series VMs like NC, NCv2, NCv3, ND, and NDv2-series VMs but works on NCsT4v3 series.

DISTRIBUTION	DRIVER
Ubuntu 16.04 LTS, 18.04 LTS, 20.04 LTS, 22.04 LTS	NVIDIA GRID 14.1, driver branch <a href="#">R510(.exe)</a>
Red Hat Enterprise Linux 7.9	
SUSE Linux Enterprise Server 15 SP2+, 15 SP2	

#### NOTE

The Azure NVads A10 v5 VMs only support GRID 14.1(510.73) or higher driver versions. Ubuntu 20.04 is not yet supported on Azure NVads A10 v5 VMs

Visit [GitHub](#) for the complete list of all previous Nvidia GRID driver links.

#### WARNING

Installation of third-party software on Red Hat products can affect the Red Hat support terms. See the [Red Hat Knowledgebase article](#).

## Install CUDA drivers on N-series VMs

Here are steps to install CUDA drivers from the NVIDIA CUDA Toolkit on N-series VMs.

C and C++ developers can optionally install the full Toolkit to build GPU-accelerated applications. For more information, see the [CUDA Installation Guide](#).

To install CUDA drivers, make an SSH connection to each VM. To verify that the system has a CUDA-capable GPU, run the following command:

```
lspci | grep -i NVIDIA
```

You will see output similar to the following example (showing an NVIDIA Tesla K80 card):

```
af8a:00:00.0 3D controller: NVIDIA Corporation GK210GL [Tesla K80] (rev a1)
```

`lspci` lists the PCIe devices on the VM, including the InfiniBand NIC and GPUs, if any. If `lspci` doesn't return successfully, you may need to install LIS on CentOS/RHEL (instructions below). Then run installation commands specific for your distribution.

#### Ubuntu

1. Download and install the CUDA drivers from the NVIDIA website.

#### NOTE

The example below shows the CUDA package path for Ubuntu 16.04. Replace the path specific to the version you plan to use.

Visit the [Nvidia Download Center](#) for the full path specific to each version.

```
CUDA_REPO_PKG=cuda-repo-ubuntu1604_10.0.130-1_amd64.deb  
wget -O /tmp/${CUDA_REPO_PKG}  
https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1604/x86\_64/\${CUDA\_REPO\_PKG}  
  
sudo dpkg -i /tmp/${CUDA_REPO_PKG}  
sudo apt-key adv --fetch-keys  
https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1604/x86\_64/3bf863cc.pub  
rm -f /tmp/${CUDA_REPO_PKG}  
  
sudo apt-get update  
sudo apt-get install cuda-drivers
```

The installation can take several minutes.

2. To optionally install the complete CUDA toolkit, type:

```
sudo apt-get install cuda
```

3. Reboot the VM and proceed to verify the installation.

#### CUDA driver updates

We recommend that you periodically update CUDA drivers after deployment.

```
sudo apt-get update  
sudo apt-get upgrade -y  
sudo apt-get dist-upgrade -y  
sudo apt-get install cuda-drivers  
  
sudo reboot
```

#### CentOS or Red Hat Enterprise Linux

1. Update the kernel (recommended). If you choose not to update the kernel, ensure that the versions of `kernel-devel` and `dkms` are appropriate for your kernel.

```
sudo yum install kernel kernel-tools kernel-headers kernel-devel  
sudo reboot
```

2. Install the latest [Linux Integration Services for Hyper-V and Azure](#). Check if LIS is required by verifying the results of `lspci`. If all GPU devices are listed as expected (and documented above), installing LIS is not required.

Please note that LIS is applicable to Red Hat Enterprise Linux, CentOS, and the Oracle Linux Red Hat Compatible Kernel 5.2-5.11, 6.0-6.10, and 7.0-7.7. Please refer to the [Linux Integration Services documentation](#) for more details. Skip this step if you plan to use CentOS/RHEL 7.8 (or higher versions) as LIS is no longer required for these versions.

```
wget https://aka.ms/lis  
tar xvzf lis  
cd LISISO  
  
sudo ./install.sh  
sudo reboot
```

3. Reconnect to the VM and continue installation with the following commands:

```
sudo rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm  
sudo yum install dkms  
  
sudo wget https://developer.download.nvidia.com/compute/cuda/repos/rhel7/x86_64/cuda-rhel7.repo -O  
/etc/yum.repos.d/cuda-rhel7.repo  
  
sudo yum install cuda-drivers
```

The installation can take several minutes.

**NOTE**

Visit [Fedora](#) and [Nvidia CUDA repo](#) to pick the correct package for the CentOS or RHEL version you want to use.

For example, CentOS 8 and RHEL 8 will need the following steps.

```
sudo rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm  
sudo yum install dkms  
  
sudo wget https://developer.download.nvidia.com/compute/cuda/repos/rhel8/x86_64/cuda-rhel8.repo -O  
/etc/yum.repos.d/cuda-rhel8.repo  
  
sudo yum install cuda-drivers
```

4. To optionally install the complete CUDA toolkit, type:

```
sudo yum install cuda
```

**NOTE**

If you see an error message related to missing packages like vulkan-filesystem then you may need to edit /etc/yum.repos.d/rh-cloud , look for optional-rpms and set enabled to 1

5. Reboot the VM and proceed to verify the installation.

### Verify driver installation

To query the GPU device state, SSH to the VM and run the [nvidia-smi](#) command-line utility installed with the driver.

If the driver is installed, you will see output similar to the following. Note that GPU-Util shows 0% unless you are currently running a GPU workload on the VM. Your driver version and GPU details may be different from the ones shown.

```

Tue Oct 10 20:48:53 2017
+-----+
| NVIDIA-SMI 384.81                    Driver Version: 384.81 |
+-----+
| GPU  Name      Persistence-M | Bus-Id     Disp.A  Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|=====+=====+=====+=====+=====+=====+=====+=====|
| 0   Tesla K80        off    | 000007D1:00:00.0 off   |      0%       0          Default |
| N/A   51C     P0    58W / 149W |        0MiB / 11439MiB |             |
+-----+
+
| Processes:                               GPU Memory |
| GPU     PID  Type  Process name        Usage  |
| =====+=====+=====+=====
| No running processes found
+-----+

```

## RDMA network connectivity

RDMA network connectivity can be enabled on RDMA-capable N-series VMs such as NC24r deployed in the same availability set or in a single placement group in a virtual machine (VM) scale set. The RDMA network supports Message Passing Interface (MPI) traffic for applications running with Intel MPI 5.x or a later version. Additional requirements follow:

### Distributions

Deploy RDMA-capable N-series VMs from one of the images in the Azure Marketplace that supports RDMA connectivity on N-series VMs:

- **Ubuntu 16.04 LTS** - Configure RDMA drivers on the VM and register with Intel to download Intel MPI:

1. Install dapl, rdmacm, ibverbs, and mlx4

```

sudo apt-get update

sudo apt-get install libdap12 libmlx4-1

```

2. In /etc/waagent.conf, enable RDMA by uncommenting the following configuration lines. You need root access to edit this file.

```

OS.EnableRDMA=y

OS.UpdateRdmaDriver=y

```

3. Add or change the following memory settings in KB in the /etc/security/limits.conf file. You need root access to edit this file. For testing purposes you can set memlock to unlimited. For example:

```
<User or group name> hard memlock unlimited.
```

```

<User or group name> hard    memlock <memory required for your application in KB>

<User or group name> soft    memlock <memory required for your application in KB>

```

4. Install Intel MPI Library. Either [purchase and download](#) the library from Intel or download the [free evaluation version](#).

```
wget http://registrationcenter-download.intel.com/akdlm/irc_nas/tec/9278/l_mpi_p_5.1.3.223.tgz
```

Only Intel MPI 5.x runtimes are supported.

For installation steps, see the [Intel MPI Library Installation Guide](#).

5. Enable ptrace for non-root non-debugger processes (needed for the most recent versions of Intel MPI).

```
echo 0 | sudo tee /proc/sys/kernel/yama/ptrace_scope
```

- **CentOS-based 7.4 HPC** - RDMA drivers and Intel MPI 5.1 are installed on the VM.
- **CentOS-based HPC** - CentOS-HPC 7.6 and later (for SKUs where InfiniBand is supported over SR-IOV). These images have Mellanox OFED and MPI libraries pre-installed.

#### NOTE

CX3-Pro cards are supported only through LTS versions of Mellanox OFED. Use LTS Mellanox OFED version (4.9-0.1.7.0) on the N-series VMs with ConnectX3-Pro cards. For more information, see [Linux Drivers](#).

Also, some of the latest Azure Marketplace HPC images have Mellanox OFED 5.1 and later, which don't support ConnectX3-Pro cards. Check the Mellanox OFED version in the HPC image before using it on VMs with ConnectX3-Pro cards.

The following images are the latest CentOS-HPC images that support ConnectX3-Pro cards:

- OpenLogic:CentOS-HPC:7.6:7.6.2020062900
- OpenLogic:CentOS-HPC:7\_6gen2:7.6.2020062901
- OpenLogic:CentOS-HPC:7.7:7.7.2020062600
- OpenLogic:CentOS-HPC:7\_7-gen2:7.7.2020062601
- OpenLogic:CentOS-HPC:8\_1:8.1.2020062400
- OpenLogic:CentOS-HPC:8\_1-gen2:8.1.2020062401

## Install GRID drivers on NV or NVv3-series VMs

To install NVIDIA GRID drivers on NV or NVv3-series VMs, make an SSH connection to each VM and follow the steps for your Linux distribution.

#### Ubuntu

1. Run the `lspci` command. Verify that the NVIDIA M60 card or cards are visible as PCI devices.
2. Install updates.

```
sudo apt-get update
sudo apt-get upgrade -y
sudo apt-get dist-upgrade -y
sudo apt-get install build-essential ubuntu-desktop -y
sudo apt-get install linux-azure -y
```

3. Disable the Nouveau kernel driver, which is incompatible with the NVIDIA driver. (Only use the NVIDIA driver on NV or NVv2 VMs.) To do this, create a file in `/etc/modprobe.d` named `nouveau.conf` with the following contents:

```
blacklist nouveau
blacklist lbm-nouveau
```

4. Reboot the VM and reconnect. Exit X server:

```
sudo systemctl stop lightdm.service
```

5. Download and install the GRID driver:

```
wget -O NVIDIA-Linux-x86_64-grid.run https://go.microsoft.com/fwlink/?LinkId=874272  
chmod +x NVIDIA-Linux-x86_64-grid.run  
sudo ./NVIDIA-Linux-x86_64-grid.run
```

6. When you're asked whether you want to run the nvidia-xconfig utility to update your X configuration file, select **Yes**.

7. After installation completes, copy /etc/nvidia/gridd.conf.template to a new file gridd.conf at location /etc/nvidia/

```
sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
```

8. Add the following to `/etc/nvidia/gridd.conf`:

```
IgnoreSP=False  
EnableUI=False
```

9. Remove the following from `/etc/nvidia/gridd.conf` if it is present:

```
FeatureType=0
```

10. Reboot the VM and proceed to verify the installation.

### **CentOS or Red Hat Enterprise Linux**

1. Update the kernel and DKMS (recommended). If you choose not to update the kernel, ensure that the versions of `kernel-devel` and `dkms` are appropriate for your kernel.

```
sudo yum update  
sudo yum install kernel-devel  
sudo rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm  
sudo yum install dkms  
sudo yum install hyperv-daemons
```

2. Disable the Nouveau kernel driver, which is incompatible with the NVIDIA driver. (Only use the NVIDIA driver on NV or NV3 VMs.) To do this, create a file in `/etc/modprobe.d` named `nouveau.conf` with the following contents:

```
blacklist nouveau  
blacklist lbm-nouveau
```

3. Reboot the VM, reconnect, and install the latest [Linux Integration Services for Hyper-V and Azure](#). Check if LIS is required by verifying the results of `lspci`. If all GPU devices are listed as expected (and documented above), installing LIS is not required.

Skip this step if you plan to use CentOS/RHEL 7.8 (or higher versions) as LIS is no longer required for these versions.

```
wget https://aka.ms/lis
tar xvzf lis
cd LISISO

sudo ./install.sh
sudo reboot
```

4. Reconnect to the VM and run the `lspci` command. Verify that the NVIDIA M60 card or cards are visible as PCI devices.

5. Download and install the GRID driver:

```
wget -O NVIDIA-Linux-x86_64-grid.run https://go.microsoft.com/fwlink/?linkid=874272
chmod +x NVIDIA-Linux-x86_64-grid.run

sudo ./NVIDIA-Linux-x86_64-grid.run
```

6. When you're asked whether you want to run the nvidia-xconfig utility to update your X configuration file, select **Yes**.

7. After installation completes, copy `/etc/nvidia/gridd.conf.template` to a new file `gridd.conf` at location `/etc/nvidia/`

```
sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
```

8. Add the following to `/etc/nvidia/gridd.conf`:

```
IgnoreSP=False
EnableUI=False
```

9. Remove the following from `/etc/nvidia/gridd.conf` if it is present:

```
FeatureType=0
```

10. Reboot the VM and proceed to verify the installation.

### Verify driver installation

To query the GPU device state, SSH to the VM and run the `nvidia-smi` command-line utility installed with the driver.

If the driver is installed, you will see output similar to the following. Note that **GPU-Util** shows 0% unless you are currently running a GPU workload on the VM. Your driver version and GPU details may be different from the ones shown.

```
[azureuser@dan lepnvr3 nvidia]$ nvidia-smi
Wed May 24 00:19:43 2017
+-----+
| NVIDIA-SMI 367.92                    Driver Version: 367.92 |
+-----+
| GPU  Name      Persistence-M | Bus-Id     Disp.A  | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap | Memory-Usage | GPU-Util Compute M. |
|=====+=====+=====+=====+=====+=====+=====+=====|
| 0   Tesla M60        off    | 8342:00:00.0  Off    | 0%          off      |
| N/A   31C   P0    39W / 150W | 0MiB /  8123MiB |           Default |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU     PID  Type  Process name        Usage      |
|=====+=====+=====+=====
| No running processes found            |
+-----+
```

## X11 server

If you need an X11 server for remote connections to an NV or NVv2 VM, [x11vnc](#) is recommended because it allows hardware acceleration of graphics. The BusID of the M60 device must be manually added to the X11 configuration file (usually, `/etc/X11/xorg.conf`). Add a `"Device"` section similar to the following:

```
Section "Device"
    Identifier      "Device0"
    Driver         "nvidia"
    VendorName    "NVIDIA Corporation"
    BoardName     "Tesla M60"
    BusID          "PCI:0@your-BusID:0:0"
EndSection
```

Additionally, update your `"Screen"` section to use this device.

The decimal BusID can be found by running

```
nvidia-xconfig --query-gpu-info | awk '/PCI BusID/{print $4}'
```

The BusID can change when a VM gets reallocated or rebooted. Therefore, you may want to create a script to update the BusID in the X11 configuration when a VM is rebooted. For example, create a script named `busidupdate.sh` (or another name you choose) with contents similar to the following:

```
#!/bin/bash
XCONFIG="/etc/X11/xorg.conf"
OLDBUSID=`awk '/BusID/{gsub(/\//, "", $2); print $2}' ${XCONFIG}`
NEWBUSID=`nvidia-xconfig --query-gpu-info | awk '/PCI BusID/{print $4}`

if [[ "${OLDBUSID}" == "${NEWBUSID}" ]] ; then
    echo "NVIDIA BusID not changed - nothing to do"
else
    echo "NVIDIA BusID changed from \"${OLDBUSID}\" to \"${NEWBUSID}\": Updating ${XCONFIG}"
    sed -e 's|BusID.*|BusID      '\"${NEWBUSID}''|' -i ${XCONFIG}
fi
```

Then, create an entry for your update script in `/etc/rc.d/rc3.d` so the script is invoked as root on boot.

## Troubleshooting

- You can set persistence mode using `nvidia-smi` so the output of the command is faster when you need to query cards. To set persistence mode, execute `nvidia-smi -pm 1`. Note that if the VM is restarted, the mode setting goes away. You can always script the mode setting to execute upon startup.

- If you updated the NVIDIA CUDA drivers to the latest version and find RDMA connectivity is no longer working, [reinstall the RDMA drivers](#) to reestablish that connectivity.
- During installation of LIS, if a certain CentOS/RHEL OS version (or kernel) is not supported for LIS, an error “Unsupported kernel version” is thrown. Please report this error along with the OS and kernel versions.
- If jobs are interrupted by ECC errors on the GPU (either correctable or uncorrectable), first check to see if the GPU meets any of Nvidia's [RMA criteria for ECC errors](#). If the GPU is eligible for RMA, please contact support about getting it serviced; otherwise, reboot your VM to reattach the GPU as described [here](#). Note that less invasive methods such as `nvidia-smi -r` do not work with the virtualization solution deployed in Azure.

## Next steps

- To capture a Linux VM image with your installed NVIDIA drivers, see [How to generalize and capture a Linux virtual machine](#).

# Install NVIDIA GPU drivers on N-series VMs running Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

To take advantage of the GPU capabilities of Azure N-series VMs backed by NVIDIA GPUs, you must install NVIDIA GPU drivers. The [NVIDIA GPU Driver Extension](#) installs appropriate NVIDIA CUDA or GRID drivers on an N-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [NVIDIA GPU Driver Extension documentation](#) for supported operating systems and deployment steps.

If you choose to install NVIDIA GPU drivers manually, this article provides supported operating systems, drivers, and installation and verification steps. Manual driver setup information is also available for [Linux VMs](#).

For basic specs, storage capacities, and disk details, see [GPU Windows VM sizes](#).

## Supported operating systems and drivers

### NVIDIA Tesla (CUDA) drivers

NVIDIA Tesla (CUDA) drivers for NC, NCv2, NCv3, NCasT4\_v3, ND, and NDv2-series VMs (optional for NV-series) are tested on the operating systems listed in the following table. CUDA driver is generic and not Azure specific. For the latest drivers, visit the [NVIDIA](#) website.

#### TIP

As an alternative to manual CUDA driver installation on a Windows Server VM, you can deploy an Azure [Data Science Virtual Machine](#) image. The DSVM editions for Windows Server 2016 pre-install NVIDIA CUDA drivers, the CUDA Deep Neural Network Library, and other tools.

OS	Driver
Windows Server 2019	<a href="#">451.82 (.exe)</a>
Windows Server 2016	<a href="#">451.82 (.exe)</a>

### NVIDIA GRID drivers

Microsoft redistributes NVIDIA GRID driver installers for NV and NVv3-series VMs used as virtual workstations or for virtual applications. Install only these GRID drivers on Azure NV-series VMs, only on the operating systems listed in the following table. These drivers include licensing for GRID Virtual GPU Software in Azure. You do not need to set up a NVIDIA vGPU software license server.

The GRID drivers redistributed by Azure do not work on non-NV series VMs like NCv2, NCv3, ND, and NDv2-series VMs. The one exception is the NCas\_T4\_V3 VM series where the GRID drivers will enable the graphics functionalities similar to NV-series.

The NC-Series with Nvidia K80 GPUs do not support GRID/graphics applications.

The Nvidia extension always installs the latest driver. The following links to previous versions are provided to support dependencies on older driver versions.

For Windows Server 2022, Windows Server 2019, Windows Server 2016 1607, 1709, Windows 10 and Windows 11:

- [GRID 14.1 \(512.78\) \(.exe\)](#)
- [GRID 13.1 \(472.39\) \(.exe\)](#)

For Windows Server 2012 R2:

- [GRID 13.1 \(472.39\) \(.exe\)](#)
- [GRID 13 \(471.68\) \(.exe\)](#)

**NOTE**

The Azure NVads A10 v5 VMs only support GRID 14.1(512.78) or higher driver versions.

For links to all previous Nvidia GRID driver versions, visit [GitHub](#).

## Driver installation

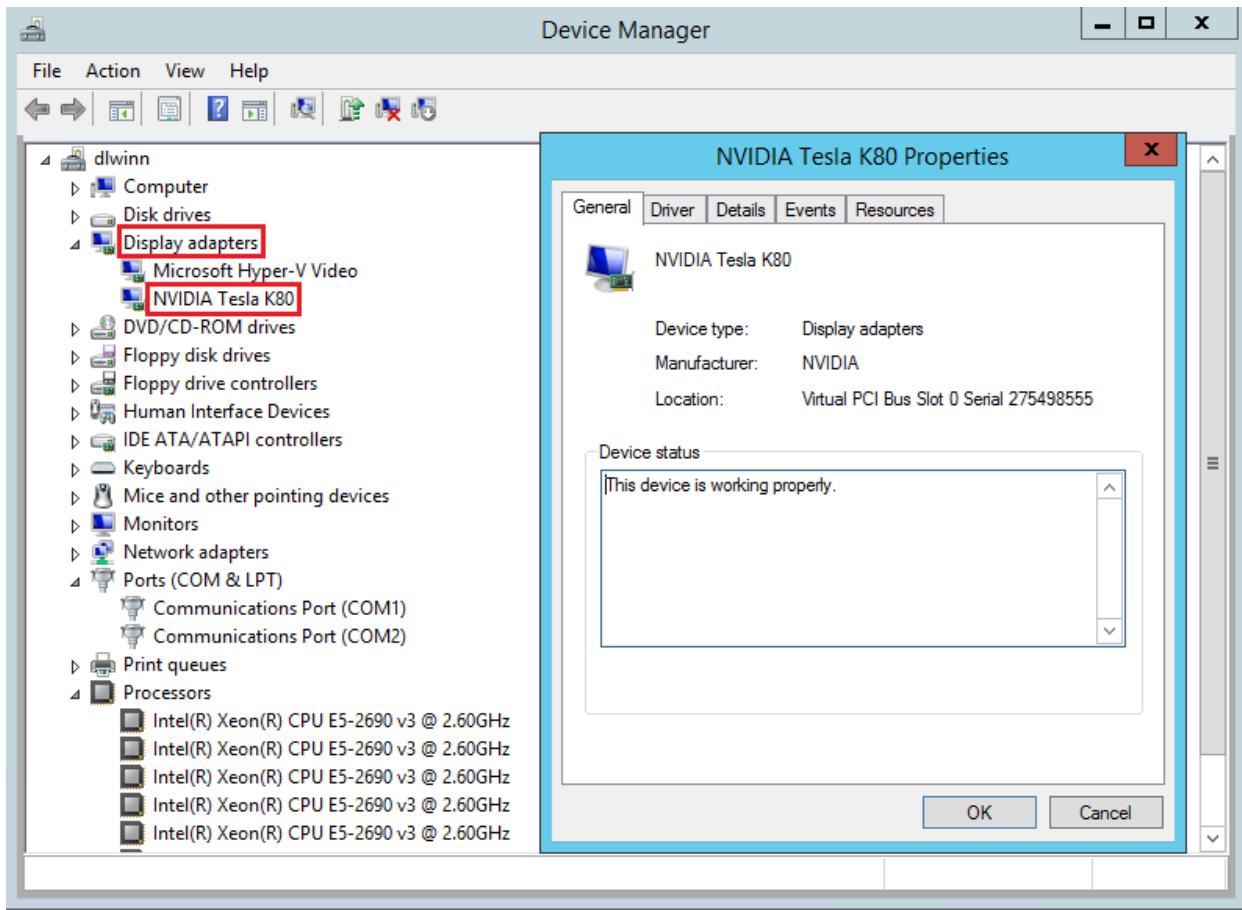
1. Connect by Remote Desktop to each N-series VM.
2. Download, extract, and install the supported driver for your Windows operating system.

After GRID driver installation on a VM, a restart is required. After CUDA driver installation, a restart is not required.

## Verify driver installation

Please note that the Nvidia Control panel is only accessible with the GRID driver installation. If you have installed CUDA drivers then the Nvidia control panel will not be visible.

You can verify driver installation in Device Manager. The following example shows successful configuration of the Tesla K80 card on an Azure NC VM.



To query the GPU device state, run the `nvidia-smi` command-line utility installed with the driver.

1. Open a command prompt and change to the `C:\Program Files\NVIDIA Corporation\NVSMI` directory.
2. Run `nvidia-smi`. If the driver is installed, you will see output similar to the following. The **GPU-Util** shows 0% unless you are currently running a GPU workload on the VM. Your driver version and GPU details may be different from the ones shown.

```
C:\Program Files\NVIDIA Corporation\NVSMI>nvidia-smi
Wed Nov 23 20:49:33 2016
+-----+
| NVIDIA-SMI 369.73       Driver Version: 369.73 |
+-----+
| GPU  Name        TCC/WDDM | Bus-Id      Disp.A  | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap| Memory-Usage | GPU-Util  Compute M. |
|-----+
|   0  Tesla K80        TCC | B794:00:00.0    Off  |          0%          Default |
| N/A   56C   P8    28W / 149W |     0MiB / 11423MiB |          0%          Default |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  PID  Type  Process name             Usage      |
|-----+
| No running processes found               0MiB      |
+-----+
```

## RDMA network connectivity

RDMA network connectivity can be enabled on RDMA-capable N-series VMs such as NC24r deployed in the same availability set or in a single placement group in a virtual machine scale set. The HpcVmDrivers extension must be added to install Windows network device drivers that enable RDMA connectivity. To add the VM extension to an RDMA-enabled N-series VM, use [Azure PowerShell](#) cmdlets for Azure Resource Manager.

To install the latest version 1.1 HpcVMDrivers extension on an existing RDMA-capable VM named myVM in the West US region:

```
Set-AzVMExtension -ResourceGroupName "myResourceGroup" -Location "westus" -VMName "myVM" -ExtensionName "HpcVmDrivers" -Publisher "Microsoft.HpcCompute" -Type "HpcVmDrivers" -TypeHandlerVersion "1.1"
```

For more information, see [Virtual machine extensions and features for Windows](#).

The RDMA network supports Message Passing Interface (MPI) traffic for applications running with [Microsoft MPI](#) or Intel MPI 5.x.

## Next steps

- Developers building GPU-accelerated applications for the NVIDIA Tesla GPUs can also download and install the latest [CUDA Toolkit](#). For more information, see the [CUDA Installation Guide](#).

# Install AMD GPU drivers on N-series VMs running Windows

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: Windows VMs ✓ Flexible scale sets

To take advantage of the GPU capabilities of the new Azure NVv4 series VMs running Windows, AMD GPU drivers must be installed. The [AMD GPU Driver Extension](#) installs AMD GPU drivers on a NVv4-series VM. Install or manage the extension using the Azure portal or tools such as Azure PowerShell or Azure Resource Manager templates. See the [AMD GPU Driver Extension documentation](#) for supported operating systems and deployment steps.

If you choose to install AMD GPU drivers manually, this article provides supported operating systems, drivers, and installation and verification steps.

Only GPU drivers published by Microsoft are supported on NVv4 VMs. Please DO NOT install GPU drivers from any other source.

For basic specs, storage capacities, and disk details, see [GPU Windows VM sizes](#).

## Supported operating systems and drivers

os	DRIVER
Windows 10 - Build 2009, 2004, 1909	<a href="#">21.Q2-1 (.exe)</a>
Windows 10 Enterprise multi-session - Build 2009, 2004, 1909	
Windows Server 2016 (version 1607)	
Windows Server 2019 (version 1909)	

Previous supported driver version for Windows builds up to 1909 is [20.Q4-1 \(.exe\)](#)

### NOTE

If you use build 1903/1909 then you may need to update the following group policy for optimal performance. These changes are not needed for any other Windows builds.

[Computer Configuration->Policies->Windows Settings->Administrative Templates->Windows Components->Remote Desktop Services->Remote Desktop Session Host->Remote Session Environment], set the Policy [Use WDDM graphics display driver for Remote Desktop Connections] to Disabled.

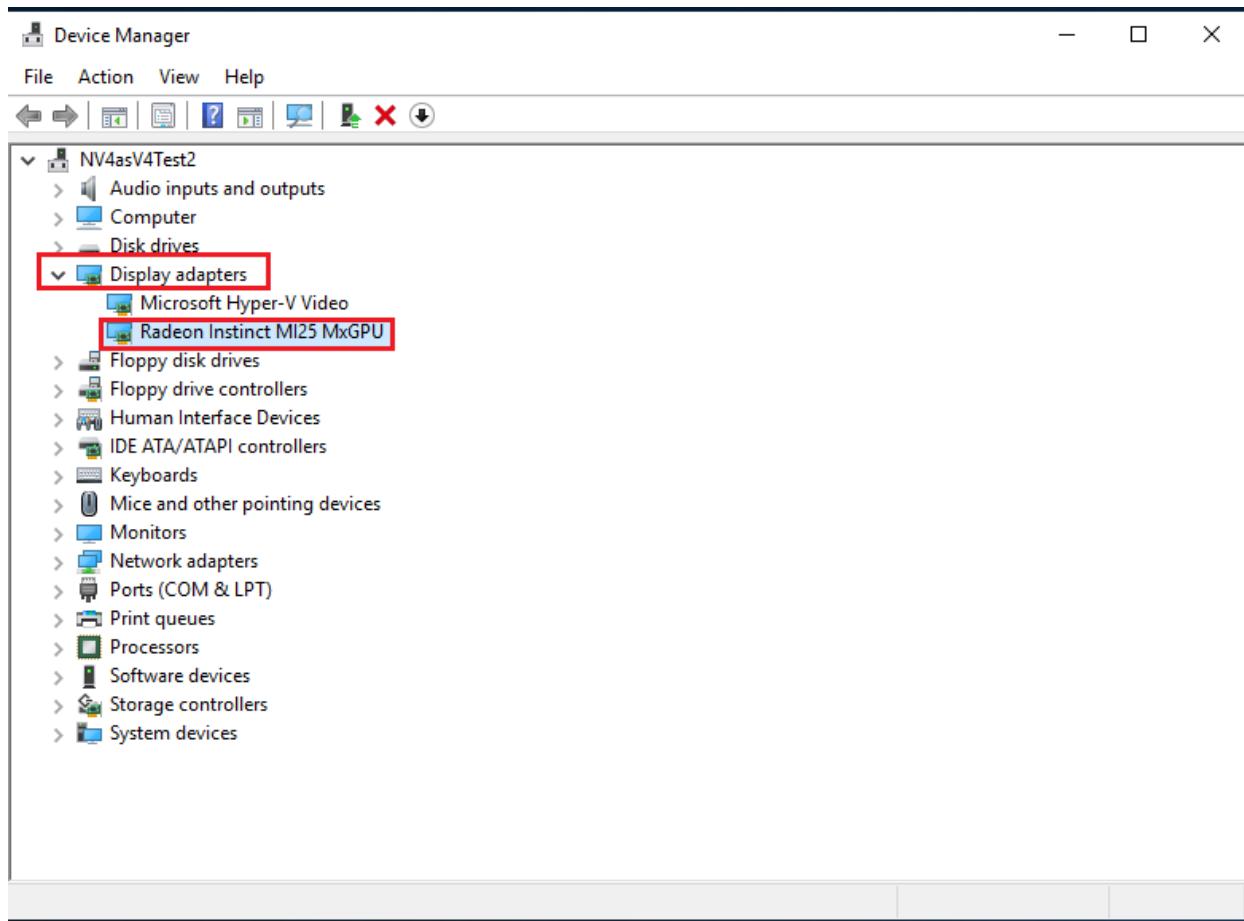
## Driver installation

1. Connect by Remote Desktop to each NVv4-series VM.
2. If you need to uninstall the previous driver version then download the [AMD cleanup utility](#). Please do not use the utility that comes with the previous version of the driver.
3. Download and install the latest driver.

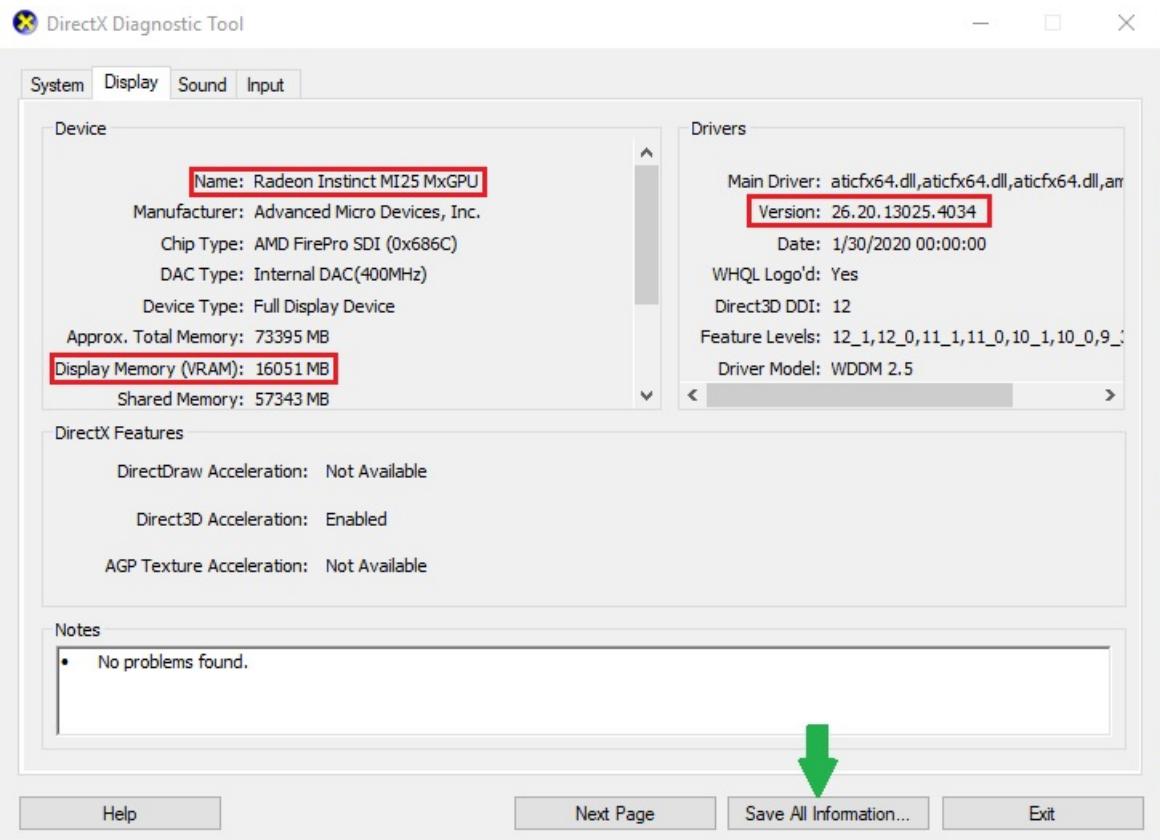
4. Reboot the VM.

## Verify driver installation

You can verify driver installation in Device Manager. The following example shows successful configuration of the Radeon Instinct MI25 card on an Azure NVv4 VM.



You can use dxdiag to verify the GPU display properties including the video RAM. The following example shows a 1/2 partition of the Radeon Instinct MI25 card on an Azure NVv4 VM.



If you are running Windows 10 build 1903 or higher then dxdiag will show no information in the 'Display' tab. Please use the 'Save All Information' option at the bottom and the output file will show the information related to AMD MI25 GPU.

-----  
Display Devices  
-----

Card name: Radeon Instinct MI25 MxGPU  
Manufacturer: Advanced Micro Devices, Inc.  
Chip type: AMD FirePro SDI (0x686C)  
DAC type: Internal DAC(400MHz)  
Device Type: Full Device  
Device Key: Enum\PCI\VEN\_1002&DEV\_686C&SUBSYS\_0C351002&REV\_00  
Device Status: 0180600A [DN\_DRIVER\_LOADED|DN\_STARTED|DN\_DISABLEABLE|DN\_REMOVABLE|DN\_NT\_ENUMERATOR|DN\_NT\_DRIVER]  
Device Problem Code: No Problem  
Driver Problem Code: Unknown  
Display Memory: 73395 MB  
Dedicated Memory: 16051 MB  
Shared Memory: 57343 MB  
Current Mode: 1920 x 1080 (32 bit) (32Hz)  
HDR Support: Unknown  
Display Topology: Unknown  
Display Color Space: DXGI\_COLOR\_SPACE\_RGB\_FULL\_G22\_NONE\_P709  
Color Primaries: Red(0.000000,0.000000), Green(0.000000,0.000000), Blue(0.000000,0.000000), White Point(0.000000,0.000000)  
Display Luminance: Min Luminance = 0.000000, Max Luminance = 0.000000, MaxFullFrameLuminance = 0.000000  
Driver Name: C:\windows\System32\DriverStore\FileRepository\c0351510.inf\_amd64\_072efa16e4548b4c\B351427\aticfx64.dll,  
C:\windows\System32\DriverStore\FileRepository\c0351510.inf\_amd64\_072efa16e4548b4c\B351427\aticfx64.dll,  
C:\windows\System32\DriverStore\FileRepository\c0351510.inf\_amd64\_072efa16e4548b4c\B351427\amdx64.dll  
**Driver File Version: 26.20.13025.4034 (English)**  
Driver Version: 26.20.13025.4034  
DDI Version: 12  
Feature Levels: 12\_1,12\_0,11\_1,11\_0,10\_1,10\_0,9\_3,9\_2,9\_1  
Driver Model: WDDM 2.5  
Graphics Preemption: Primitive  
Compute Preemption: DMA  
Miracast: Not Supported  
Hybrid Graphics GPU: Not Supported  
Power P-states: Not Supported  
Virtualization: Not Supported  
Block List: GPU\_PV\_HIGH\_SECURITY  
Catalog Attributes: Universal:False Declarative:False  
Driver Attributes: Final Retail  
**Driver Date/Size: 1/30/2020 12:00:00 AM, 1979552 bytes**  
WHQL Logo'd: Yes  
WHQL Date Stamp: Unknown  
Device Identifier: {D7B71EE2-2B2C-11CF-707C-D41EBBC2D735}  
Vendor ID: 0x1002  
Device ID: 0x686C  
SubSys ID: 0x0C351002  
Revision ID: 0x0000  
Driver Strong Name: oem2.inf:cb0ae414494e7940:ati2mtag\_R7500DS:26.20.13025.4034:pci\ven\_1002&dev\_686c&rev\_00  
Rank Of Driver: 00D12000  
Video Accel: Unknown  
DXVA2 Modes:  
Deinterlace Caps: n/a  
D3D9 Overlay: Not Supported  
DXVA-HD: Not Supported  
DDraw Status: Not Available  
D3D Status: Enabled  
AGP Status: Not Available  
MPO MaxPlanes: 1  
MPO Caps: Not Supported  
MPO Stretch: Not Supported  
MPO Media Hints: Not Supported  
MPO Formats: Not Supported  
PanelFitter Caps: Not Supported  
PanelFitter Stretch: Not Supported

# Migrate your NC and NC\_Promo series virtual machines by August 31, 2023

9/21/2022 • 2 minutes to read • [Edit Online](#)

Based on feedback we've received from customers we're happy to announce that we are extending the retirement date by 1 year to 31 August 2023, for the Azure NC-Series virtual machine to give you more time to plan your migration.

As we continue to bring modern and optimized virtual machine instances to Azure leveraging the latest innovations in datacenter technologies, we thoughtfully plan how we retire aging hardware. With this in mind, we are retiring our NC (v1) GPU VM sizes, powered by NVIDIA Tesla K80 GPUs on 31 August 2023.

## How does the NC-series migration affect me?

After 31 August 2023, any remaining NC size virtual machines remaining in your subscription will be set to a deallocated state. These virtual machines will be stopped and removed from the host. These virtual machines will no longer be billed in the deallocated state.

This VM size retirement only impacts the VM sizes in the [NC-series](#). This does not impact the newer [NCv3](#), [NC T4 v3](#), and [ND v2](#) series virtual machines.

## What actions should I take?

You will need to resize or deallocate your NC virtual machines. We recommend moving your GPU workloads to another GPU Virtual Machine size. Learn more about migrating your workloads to another [GPU Accelerated Virtual Machine size](#).

## Next steps

[Learn more](#) about migrating your workloads to other GPU Azure Virtual Machine sizes.

If you have questions, contact us through customer support.

# Migrate your NCv2 series virtual machines by August 31, 2023

9/21/2022 • 2 minutes to read • [Edit Online](#)

Based on feedback we've received from customers we're happy to announce that we are extending the retirement date by 1 year to August 31, 2023, for the Azure NCv2-Series virtual machine to give you more time to plan your migration.

As we continue to bring modern and optimized virtual machine instances to Azure leveraging the latest innovations in datacenter technologies, we thoughtfully plan how we retire aging hardware. With this in mind, we are retiring our NC (v2) GPU VM sizes, powered by NVIDIA Tesla P100 GPUs on 31 August 2023.

## How does the NCv2-series migration affect me?

After 31 August 2023, any remaining NCv2 size virtual machines remaining in your subscription will be set to a deallocated state. These virtual machines will be stopped and removed from the host. These virtual machines will no longer be billed in the deallocated state.

This VM size retirement only impacts the VM sizes in the [NCv2-series](#). This does not impact the newer [NCv3](#), [NC T4 v3](#), and [ND v2](#) series virtual machines.

## What actions should I take?

You will need to resize or deallocate your NC virtual machines. We recommend moving your GPU workloads to another GPU Virtual Machine size. Learn more about migrating your workloads to another [GPU Accelerated Virtual Machine size](#).

## Next steps

[Learn more](#) about migrating your workloads to other GPU Azure Virtual Machine sizes.

If you have questions, contact us through customer support.

# Migrate your ND series virtual machines by August 31, 2023

9/21/2022 • 2 minutes to read • [Edit Online](#)

Based on feedback we've received from customers we're happy to announce that we are extending the retirement date by 1 year to 31 August 2023, for the Azure ND-Series virtual machine to give you more time to plan your migration.

As we continue to bring modern and optimized virtual machine instances to Azure leveraging the latest innovations in datacenter technologies, we thoughtfully plan how we retire aging hardware. With this in mind, we are retiring our ND GPU VM sizes, powered by NVIDIA Tesla P40 GPUs on 31 August 2023.

## How does the ND-series migration affect me?

After 31 August 2023, any remaining ND size virtual machines remaining in your subscription will be set to a deallocated state. These virtual machines will be stopped and removed from the host. These virtual machines will no longer be billed in the deallocated state.

This VM size retirement only impacts the VM sizes in the **ND-series**. This does not impact the newer [NCv3](#), [NC T4 v3](#), and [ND v2](#) series virtual machines.

## What actions should I take?

You will need to resize or deallocate your ND virtual machines. We recommend moving your GPU workloads to another GPU Virtual Machine size. Learn more about migrating your workloads to another [GPU Accelerated Virtual Machine size](#).

## Next steps

[Learn more](#) about migrating your workloads to other GPU Azure Virtual Machine sizes.

If you have questions, contact us through customer support.

# Migration Guide for GPU Compute Workloads in Azure

9/21/2022 • 10 minutes to read • [Edit Online](#)

As more powerful GPUs become available in the marketplace and in Microsoft Azure datacenters, we recommend re-assessing the performance of your workloads and considering migrating to newer GPUs.

For the same reason, as well as to maintain a high-quality and reliable service offering, Azure periodically retires the hardware that powers older VM sizes. The first group of GPU products to be retired in Azure are the original NC, NC v2 and ND-series VMs, powered by NVIDIA Tesla K80, P100, and P40 datacenter GPU accelerators respectively. These products will be retired on August 31st 2023, and the oldest VMs in this series launched in 2016.

Since then, GPUs have made incredible strides alongside the entire deep learning and HPC industry, typically exceeding a doubling in performance between generations. Since the launch of NVIDIA K80, P40, and P100 GPUs, Azure has shipped multiple newer generations and categories of VM products geared at GPU-accelerated compute and AI, based around NVIDIA's T4, V100, and A100 GPUs, and differentiated by optional features such as InfiniBand-based interconnect fabrics. These are all options we encourage customers to explore as migration paths.

In most cases, the dramatic increase in performance offered by newer generations of GPUs lowers overall TCO by decreasing the duration of job, for burstable jobs- or reducing the quantity of overall GPU-enabled VMs required to cover a fixed-size demand for compute resources, even though costs per GPU-hour may vary. In addition to these benefits, customers may improve Time-to-Solution via higher-performing VMs, and improve the health and supportability of their solution by adopting newer software, CUDA runtime, and driver versions.

## Migration vs. Optimization

Azure recognizes that customers have a multitude of requirements that may dictate the selection of a specific GPU VM product, including GPU architectural considerations, interconnects, TCO, Time to Solution, and regional availability based on compliance locality or latency requirements- and some of these even change over time.

At the same time, GPU acceleration is a new and rapidly evolving area.

Thus, there is no true one-size fits-all guidance for this product area, and a migration is a perfect time to re-evaluate potentially dramatic changes to a workload- like moving from a clustered deployment model to a single large 8-GPU VM or vice versa, leveraging reduced precision datatypes, adopting features like Multi-Instance GPU, and much more.

These sorts of considerations- when made the context of already dramatic per-generation GPU performance increases, where a feature such as the addition of TensorCores can increase performance by an order of magnitude, are extremely workload-specific.

Combining migration with application re-architecture can yield immense value and improvement in cost and time-to-solution.

However, these sorts of improvements are beyond the scope of this document, which aims to focus on direct equivalency classes for generalized workloads that may be run by customers today, to identify the most similar VM options in both price *and* performance per GPU to existing VM families undergoing retirement.

Thus, this document assumes that the user may not have any insight or control over workload-specific properties like the number of required VM instances, GPUs, interconnects, and more.

# Recommended Upgrade Paths

## NC-Series VMs featuring NVIDIA K80 GPUs

The [NC \(v1\)-Series](#) VMs are Azure's oldest GPU-accelerated compute VM type, powered by 1 to 4 NVIDIA Tesla K80 datacenter GPU accelerators paired with Intel Xeon E5-2690 v3 (Haswell) processors. Once a flagship VM type for demanding AI, ML, and HPC applications, they remained a popular choice late into the product lifecycle (particularly via NC-series promotional pricing) for users who valued having a very low absolute cost per GPU-hour over GPUs with higher throughput-per-dollar.

Today, given the relatively low compute performance of the aging NVIDIA K80 GPU platform, in comparison to VM series featuring newer GPUs, a popular use case for the NC-series is real-time inference and analytics workloads, where an accelerated VM must be available in a steady state to serve request from applications as they arrive. In these cases the volume or batch size of requests may be insufficient to benefit from more performant GPUs. NC VMs are also popular for developers and students learning about, developing for, or experimenting with GPU acceleration, who need an inexpensive cloud-based CUDA deployment target upon which to iterate that doesn't need to perform to production levels.

In general, NC-Series customers should consider moving directly across from NC sizes to [NC T4 v3](#) sizes, Azure's new GPU-accelerated platform for light workloads powered by NVIDIA Tesla T4 GPUs, although other VM SKUs should be considered for workloads running on InfiniBand-enabled NC-Series sizes.

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NC6 Standard_NC6_Promo	Standard_NC4as_T4_v3 or Standard_NC8as_T4	CPU: Intel Haswell vs AMD Rome GPU count: 1 (same) GPU generation: NVIDIA Kepler vs. Turing (+2 generations, ~2x FP32 FLOPs) GPU memory (GiB per GPU): 16 (+4) vCPU: 4 (-2) or 8 (+2) Memory GiB: 16 (-40) or 56 (same) Temp Storage (SSD) GiB: 180 (-160) or 360 (+20) Max data disks: 8 (-4) or 16 (+4) Accelerated Networking: Yes (+) Premium Storage: Yes (+)
Standard_NC12 Standard_NC12_Promo	Standard_NC16as_T4_v3	CPU: Intel Haswell vs AMD Rome GPU count: 1 (-1) GPU generation: NVIDIA Kepler vs. Turing (+2 generations, ~2x FP32 FLOPs) GPU memory (GiB per GPU): 16 (+4) vCPU: 16 (+4) Memory GiB: 110 (-2) Temp Storage (SSD) GiB: 360 (-320) Max data disks: 48 (+16) Accelerated Networking: Yes (+) Premium Storage: Yes (+)

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NC24 Standard_NC24_Promo	Standard_NC64as_T4_v3*	CPU: Intel Haswell vs AMD Rome GPU count: 4 (same) GPU generation: NVIDIA Kepler vs. Turing (+2 generations, ~2x FP32 FLOPs) GPU memory (GiB per GPU): 16 (+4) vCPU: 64 (+40) Memory GiB: 440 (+216) Temp Storage (SSD) GiB: 2880 (+1440) Max data disks: 32 (-32) Accelerated Networking: Yes (+) Premium Storage: Yes (+)
Standard_NC24r Standard_NC24r_Promo  (InfiniBand clustering-enabled sizes)	Standard_NC24rs_v3*	CPU: Intel Haswell vs Intel Broadwell GPU count: 4 (same) GPU generation: NVIDIA Kepler vs. Volta (+2 generations) GPU memory (GiB per GPU): 16 (+4) vCPU: 24 (+0) Memory GiB: 448 (+224) Temp Storage (SSD) GiB: 2948 (+1440) Max data disks: 32 (same) Accelerated Networking: No (Same) Premium Storage: Yes (+) InfiniBand interconnect: Yes

### ND-Series VMs featuring NVIDIA Tesla P40 GPUs

The ND-series virtual machines are a midrange platform originally designed for AI and Deep Learning workloads. They offered excellent performance for batch inferencing via improved single-precision floating point operations over their predecessors and are powered by NVIDIA Tesla P40 GPUs and Intel Xeon E5-2690 v4 (Broadwell) CPUs. Like the NC and NC v2-Series, the ND-Series offers a configuration with a secondary low-latency, high-throughput network through RDMA, and InfiniBand connectivity so you can run large-scale training jobs spanning many GPUs.

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_ND6	Standard_NC4as_T4_v3 or Standard_NC8as_T4	CPU: Intel Broadwell vs AMD Rome GPU count: 1 (same) GPU generation: NVIDIA Pascal vs. Turing (+1 generation) GPU memory (GiB per GPU): 16 (-8) vCPU: 4 (-2) or 8 (+2) Memory GiB: 16 (-40) or 56 (-56) Temp Storage (SSD) GiB: 180 (-552) or 360 (-372) Max data disks: 8 (-4) or 16 (+4) Accelerated Networking: Yes (+) Premium Storage: Yes (+)

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_ND12	Standard_NC16as_T4_v3	CPU: Intel Broadwell vs AMD Rome GPU count: 1 (-1) GPU generation: NVIDIA Pascal vs. Turing (+1 generations) GPU memory (GiB per GPU): 16 (-8) vCPU: 16 (+4) Memory GiB: 110 (-114) Temp Storage (SSD) GiB: 360 (-1,114) Max data disks: 48 (+16) Accelerated Networking: Yes (+) Premium Storage: Yes (+)
Standard_ND24	Standard_NC64as_T4_v3*	CPU: Intel Broadwell vs AMD Rome GPU count: 4 (same) GPU generation: NVIDIA Pascal vs. Turing (+1 generations) GPU memory (GiB per GPU): 16 (-8) vCPU: 64 (+40) Memory GiB: 440 (same) Temp Storage (SSD) GiB: 2880 (same) Max data disks: 32 (same) Accelerated Networking: Yes (+) Premium Storage: Yes (+)
Standard_ND24r	Standard_NC24rs_v3*	CPU: Intel Broadwell (Same) GPU count: 4 (same) GPU generation: NVIDIA Pascal vs. Volta (+1 generation) GPU memory (GiB per GPU): 16 (-8) vCPU: 24 (+0) Memory GiB: 448 (same) Temp Storage (SSD) GiB: 2948 (same) Max data disks: 32 (same) Accelerated Networking: No (Same) Premium Storage: Yes (+) InfiniBand interconnect: Yes (Same)

### NC v2-Series VMs featuring NVIDIA Tesla P100 GPUs

The NC v2-series virtual machines are a flagship platform originally designed for AI and Deep Learning workloads. They offered excellent performance for Deep Learning training, with per-GPU performance roughly 2x that of the original NC-Series and are powered by NVIDIA Tesla P100 GPUs and Intel Xeon E5-2690 v4 (Broadwell) CPUs. Like the NC and ND -Series, the NC v2-Series offers a configuration with a secondary low-latency, high-throughput network through RDMA, and InfiniBand connectivity so you can run large-scale training jobs spanning many GPUs.

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
-----------------	----------------	-----------------------------

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NC6s_v2	Standard_NC6s_v3	CPU: Intel Broadwell (Same) GPU count: 1 (same) GPU generation: NVIDIA Pascal vs. Volta (+1 generation) GPU memory (GiB per GPU): 16 (same) vCPU: 6 (same) Memory GiB: 112 (same) Temp Storage (SSD) GiB: 736 (same) Max data disks: 12 (same) Accelerated Networking: No (same) Premium Storage: Yes (+)
Standard_NC12s_v2	Standard_NC12s_v3	CPU: Intel Broadwell (Same) GPU count: 2 (same) GPU generation: NVIDIA Pascal vs. Volta (+1 generations) GPU memory (GiB per GPU): 16 (same) vCPU: 12 (same) Memory GiB: 112 (same) Temp Storage (SSD) GiB: 1474 (same) Max data disks: 24 (same) Accelerated Networking: No (same) Premium Storage: Yes (+)
Standard_NC24s_v2	Standard_NC24s_v3	CPU: Intel Broadwell (same) GPU count: 4 (same) GPU generation: NVIDIA Pascal vs. Volta (+1 generations) GPU memory (GiB per GPU): 16 (same) vCPU: 24 (same) Memory GiB: 448 (same) Temp Storage (SSD) GiB: 2948 (same) Max data disks: 32 (same) Accelerated Networking: No (same) Premium Storage: Yes (+)
Standard_NC24rs_v2	Standard_NC24rs_v3*	CPU: Intel Broadwell (same) GPU count: 4 (same) GPU generation: NVIDIA Pascal vs. Volta (+1 generations) GPU memory (GiB per GPU): 16 (same) vCPU: 24 (same) Memory GiB: 448 (same) Temp Storage (SSD) GiB: 2948 (same) Max data disks: 32 (same) Accelerated Networking: No (same) Premium Storage: Yes (+) InfiniBand interconnect: Yes (Same)

## Migration Steps

### General Changes

1. Choose a series and size for migration. Leverage the [pricing calculator](#) for further insights.
2. Get quota for the target VM series

3. Resize the current N\* series VM size to the target size. This may also be a good time to update the operating system used by your Virtual Machine image, or adopt one of the HPC images with drivers pre-installed as your starting point.

#### **IMPORTANT**

Your VM image may have been produced with an older version of the CUDA runtime, NVIDIA driver, and (if applicable, for RDMA-enabled sizes only) Mellanox OFED drivers than your new GPU VM series requires, which can be updated by [following the instructions in the Azure Documentation](#).

## **Breaking Changes**

### **Select target size for migration**

After assessing your current usage, decide what type of GPU VM you need. Depending on the workload requirements you have few different choices.

#### **NOTE**

A best practice is to select a VM size based on both cost and performance. The recommendations in this guide are based on a general-purpose, one-to-one comparison of performance metrics and the nearest match in another VM series. Before deciding on the right size, get a cost comparison using the [Azure Pricing Calculator](#).

#### **IMPORTANT**

All legacy NC, NC v2 and ND-Series sizes are available in multi-GPU sizes, including 4-GPU sizes with and without InfiniBand interconnect for scale-out, tightly-coupled workloads that demand more compute power than a single 4-GPU VM, or a single K80, P40, or P100 GPU can supply respectively. Although the recommendations above offer a straightforward path forward, users of these sizes should consider achieving their performance goals with more powerful NVIDIA V100 GPU-based VM series like the [NC v3-Series](#) and [ND v2-series](#), which typically enable the same level of workload performance at lower costs and with improved manageability by providing considerably greater performance per GPU and per VM before multi-GPU and multi-node configurations are required, respectively.

### **Get quota for the target VM family**

Follow the guide to [request an increase in vCPU quota by VM family](#). Select the target VM size you have selected for migration.

### **Resize the current virtual machine**

You can [resize the virtual machine](#).

## **Next steps**

For a full list of GPU enabled virtual machine sizes, see [GPU - accelerated compute overview](#)

# Migrate your NV and NV\_Promo series virtual machines by August 31, 2023

9/21/2022 • 2 minutes to read • [Edit Online](#)

Based on feedback we've received from customers we're happy to announce that we are extending the retirement date by 1 year to August 31, 2023, for the Azure NV-Series and NV\_Promo Series virtual machine to give you more time to plan your migration.

We continue to bring modern and optimized virtual machine (VM) instances to Azure by using the latest innovations in datacenter technologies. As we innovate, we also thoughtfully plan how we retire aging hardware. With this context in mind, we're retiring our NV-series Azure VM sizes on August 31, 2023.

## How does the NV series migration affect me?

After August 31, 2023, any remaining NV and NV\_Promo-size VMs remaining in your subscription will be set to a deallocated state. These VMs will be stopped and removed from the host. These VMs will no longer be billed in the deallocated state.

The current VM size retirement only affects the VM sizes in the [NV series](#). This retirement doesn't affect the [NVv3](#) and [NVv4](#) series VMs.

## What actions should I take?

You'll need to resize or deallocate your NV VMs. We recommend moving your GPU visualizations or graphics workloads to another [GPU accelerated VM size](#).

[Learn more](#) about migrating your workloads to other GPU Azure VM sizes.

If you have questions, contact us through customer support.

# NV series migration guide

9/21/2022 • 4 minutes to read • [Edit Online](#)

As more powerful GPU VM sizes become available in Azure datacenters, assess your workloads and migrate virtual machines (VMs) in the NV and NV\_Promo series. These legacy VMs can be migrated into new VM series, such as NVsv3 and NVasv4, for better performance with reduced cost. The NVsv3 VM series is powered by Nvidia M60 GPUs. The NVasv4 series is powered by AMD Radeon Instinct MI25 GPUs.

The main differences between the NV and NV\_Promo series and the newer NVsv3 and NVasv4 series are:

- Improved performance.
- Support for Premium storage.
- The option to choose from a fractional GPU size to multi-GPU configurations.

Both the NVsv3 and NVasv4 series have more modern cores and greater capacity.

The following section summarizes the differences between the legacy NV series and the NVsv3 and NVv4 series.

## NVsv3 series

The NVv3-series VMs are powered by NVIDIA Tesla M60 GPUs and NVIDIA GRID technology with Intel E5-2690 v4 (Broadwell) CPUs and Intel Hyper-Threading Technology. These VMs are targeted for GPU accelerated graphics applications and virtual desktops where customers want to:

- Visualize their data.
- Simulate results to view.
- Work on CAD.
- Render and stream content.

These VMs can also run single precision workloads, such as encoding and rendering.

NVv3 VMs support Premium storage and come with twice the system memory (RAM) when compared with the NV series. For the most up-to-date specifications, see [GPU accelerated compute VM sizes: NVsv3 series](#).

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NV6 Standard_NV6_Promo	Standard_NV12s_v3	vCPU: 12 (+6) Memory: GiB 112 (+56) Temp storage (SSD) GiB: 320 (-20) Max data disks: 12 (-12) Accelerated networking: Yes Premium storage: Yes
Standard_NV12 Standard_NV12_Promo	Standard_NV24s_v3	vCPU: 24 (+12) Memory: GiB 224 (+112) Temp storage (SSD) GiB: 640 (-40) Max data disks: 24 (-24) Accelerated networking: Yes Premium storage: Yes

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NV24 Standard_NV24_Promo	Standard_NV48s_v3	vCPU: 48 (+24) Memory: GiB 448 (+224) Temp storage (SSD) GiB: 1280 (-160) Max data disks: 32 (-32) Accelerated networking: Yes Premium storage: Yes

## NVv4 series

NVv4-series VMs are powered by AMD Radeon Instinct MI25 GPUs and AMD EPYC 7V12 (Rome) CPUs. With the NVv4 series, Azure introduces VMs with partial GPUs. Choose the right size VM for GPU accelerated graphics applications and virtual desktops that start at one-eighth of a GPU with a 2-GiB frame buffer to a full GPU with a 16-GiB frame buffer.

NVv4 VMs currently support only the Windows guest operating system. For the most up-to-date specifications, see [GPU accelerated compute VM sizes: NVv4 series](#).

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_NV6 Standard_NV6_Promo	Standard_NV16as_v4	vCPU: 16 (+10) Memory: GiB 56 Temp storage (SSD) GiB: 352 (+12) Max data disks: 16 (-8) Accelerated networking: Yes Premium storage: Yes
Standard_NV12 Standard_NV12_Promo	Standard_NV32as_v4	vCPU: 32 (+20) Memory: GiB 112 Temp storage (SSD) GiB: 704 (+24) Max data disks: 32 (+16) Accelerated networking: Yes Premium storage: Yes
Standard_NV24 Standard_NV24_Promo	N/A	N/A

## Migration steps for general changes

To deal with general changes:

1. Choose a series and size for migration.
2. Get a quota for the target VM series.
3. Resize the current NV-series VM size to the target size.

If the target size is NVv4, make sure to remove the Nvidia GPU driver and install the AMD GPU driver.

## Migration steps for breaking changes

To deal with breaking changes, follow the steps in the next sections.

### Select a target size for migration

After you assess your current usage, decide what type of GPU VM you need. Depending on the workload requirements, you have a few different choices. Here's how to choose:

- If the workload is graphics or visualizations and has a hard dependency on using the Nvidia GPU, migrate to the NVsv3 series.
- If the workload is graphics or visualizations and has no hard dependency on a specific type of GPU, migrate to the NVsv3 or NVVasv4 series.

#### NOTE

A best practice is to select a VM size based on both cost and performance. The recommendations in this article are based on a one-to-one comparison of performance metrics for the NV and NV\_Promo sizes and the nearest match in another VM series. Before you decide on the right size, get a cost comparison by using the Azure Pricing Calculator.

### Get a quota for the target VM family

Follow the guide to [request an increase in vCPU quota by VM family](#). Select the NVSv3 series or NVv4 series as the VM family name depending on the target VM size you selected for migration.

### Resize the current VM

You can [resize the VM](#).

## FAQ

**Q:** Which GPU driver should I use for the target VM size?

**A:** For the NVsv3 series, use the [Nvidia GRID driver](#). For NVv4, use the [AMD GPU drivers](#).

**Q:** I use the Nvidia GPU driver extension today. Will it work for the target VM size?

**A:** The current [Nvidia driver extension](#) will work for NVsv3. Use the [AMD GPU driver extensions](#) if the target VM size is NVv4.

**Q:** Which target VM series should I use if I have dependency on CUDA?

**A:** NVv3 supports CUDA. The NVv4 VM series with the AMD GPUs doesn't support CUDA.

# FPGA optimized virtual machine sizes

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

FPGA optimized VM sizes are specialized virtual machines available with single or multiple FPGAs. These sizes are designed for compute-intensive workloads. This article provides information about the number and type of FPGAs, vCPUs, data disks, and NICs. Storage throughput and network bandwidth are also included for each size in this grouping.

- The [NP-series](#) sizes are optimized for workloads including machine learning inference, video transcoding, and database search & analytics. The NP-series are powered by Xilinx U250 accelerators.

## Deployment considerations

- For availability of N-series VMs, see [Products available by region](#).
- N-series VMs can only be deployed in the Resource Manager deployment model.
- If you want to deploy more than a few N-series VMs, consider a pay-as-you-go subscription or other purchase options. If you're using an [Azure free account](#), you can use only a limited number of Azure compute cores.
- You might need to increase the cores quota (per region) in your Azure subscription, and increase the separate quota for NP cores. To request a quota increase, [open an online customer support request](#) at no charge. Default limits may vary depending on your subscription category.

## Other sizes

- [General purpose](#)
- [Compute optimized](#)
- [GPU accelerated compute](#)
- [High performance compute](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [Previous generations](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# NP-series

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The NP-series virtual machines are powered by [Xilinx U250](#) FPGAs for accelerating workloads including machine learning inference, video transcoding, and database search & analytics. NP-series VMs are also powered by Intel Xeon 8171M (Skylake) CPUs with all core turbo clock speed of 3.2 GHz.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1

[Accelerated Networking](#): Supported

[Ephemeral OS Disks](#): Supported

[Nested Virtualization](#): Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	FPGA	FPGA MEMORY: GIB	MAX DATA DISKS	MAX NICs/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_N_P10s	10	168	736	1	64	8	1 / 7500
Standard_N_P20s	20	336	1474	2	128	16	2 / 15000
Standard_N_P40s	40	672	2948	4	256	32	4 / 30000

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended

application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Frequently asked questions

**Q:** How to request quota for NP VMs?

**A:** Please follow this page [Increase limits by VM series](#). NP VMs are available in East US, West US2, West Europe, SouthEast Asia, and SouthCentral US.

**Q:** What version of Vitis should I use?

**A:** Xilinx recommends [Vitis 2021.1](#), you can also use the Development VM marketplace options (Vitis 2021.1 Development VM for Ubuntu 18.04, Ubuntu 20.04, and CentOS 7.8)

**Q:** Do I need to use NP VMs to develop my solution?

**A:** No, you can develop on-premises and deploy to the cloud. Please make sure to follow the [attestation documentation](#) to deploy on NP VMs.

**Q:** Which file returned from attestation should I use when programming my FPGA in an NP VM?

**A:** Attestation returns two xclbins, **design.bit.xclbin** and **design.azure.xclbin**. Please use **design.azure.xclbin**.

**Q:** Where should I get all the XRT / Platform files?

**A:** Please visit Xilinx's [Microsoft-Azure](#) site for all files.

**Q:** What Version of XRT should I use?

**A:** xrt\_202110.2.11.680

**Q:** What is the target deployment platform?

**A:** Use the following platforms.

- xilinx-u250-gen3x16-xdma-platform-2.1-3\_all
- xilinx-u250-gen3x16-xdma-validate\_2.1-3005608.1

**Q:** Which platform should I target for development?

**A:** xilinx-u250-gen3x16-xdma-2.1-202010-1-dev\_1-2954688\_all

**Q:** What are the supported Operating Systems?

**A:** Xilinx and Microsoft have validated Ubuntu 18.04 LTS, Ubuntu 20.04 LTS, and CentOS 7.8.

Xilinx has created the following marketplace images to simplify the deployment of these VMs.

Xilinx Alveo U250 2021.1 Deployment VM – Ubuntu18.04

Xilinx Alveo U250 2021.1 Deployment VM – Ubuntu20.04

Xilinx Alveo U250 2021.1 Deployment VM – CentOS7.8

**Q:** Can I deploy my Own Ubuntu / CentOS VMs and install XRT / Deployment Target Platform?

**A:** Yes.

Q: If I deploy my own Ubuntu18.04 VM then what are the required packages and steps?

A: Use Kernel 4.15 per [Xilinx XRT documentation](#)

Install the following packages.

- xrt\_202110.2.11.680\_18.04-amd64-xrt.deb
- xrt\_202110.2.11.680\_18.04-amd64-azure.deb
- xilinx-u250-gen3x16-xdma-platform-2.1-3\_all\_18.04.deb.tar.gz
- xilinx-u250-gen3x16-xdma-validate\_2.1-3005608.1\_all.deb

Q: On Ubuntu, after rebooting my VM I can't find my FPGA(s):

A: Please verify that your kernel hasn't been upgraded (uname -a). If so, please downgrade to kernel 4.1X.

Q: If I deploy my own Ubuntu20.04 VM then what are the required packages and steps?

A: Use Kernel 5.4 per [Xilinx XRT documentation](#)

Install the following packages.

- xrt\_202110.2.11.680\_20.04-amd64-xrt.deb
- xrt\_202110.2.11.680\_20.04-amd64-azure.deb
- xilinx-u250-gen3x16-xdma-platform-2.1-3\_all\_18.04.deb.tar.gz
- xilinx-u250-gen3x16-xdma-validate\_2.1-3005608.1\_all.deb

Q: If I deploy my own CentOS7.8 VM then what are the required packages and steps?

A: Use Kernel version: 3.10.0-1160.15.2.el7.x86\_64

Install the following packages.

- xrt\_202110.2.11.680\_7.8.2003-x86\_64-xrt.rpm
- xrt\_202110.2.11.680\_7.8.2003-x86\_64-azure.rpm
- xilinx-u250-gen3x16-xdma-platform-2.1-3.noarch.rpm.tar.gz
- xilinx-u250-gen3x16-xdma-validate-2.1-3005608.1.noarch.rpm

Q: What are the differences between OnPrem and NP VMs?

A:

- **Regarding XOCL/XCLMGMT:**

On Azure NP VMs, only the role endpoint (Device ID 5005), which uses the XOCL driver, is present.

OnPrem FPGA, both the management endpoint (Device ID 5004) and role endpoint (Device ID 5005), which use the XCLMGMT and XOCL drivers respectively, are present.

- **Regarding XRT:**

On Azure NP VMs, the XDMA 2.1 platform only supports Host\_Mem(SB) and DDR data retention features.

To enable Host\_Mem(SB) (up to 1 Gb RAM): sudo xbutil host\_mem --enable --size 1g

To disable Host\_Mem(SB): sudo xbutil host\_mem --disable

Starting on XRT2021.1:

OnPrem FPGA in Linux exposes [M2M data transfer](#).

This feature is not supported in Azure NP VMs.

Q: Can I run xbmgmt commands?

A: No, on Azure VMs there's no management support directly from the Azure VM.

Q: Do I need to load a PLP?

A: No, the PLP is loaded automatically for you, so there's no need to load via xbmgmt commands.

Q: Does Azure support different PLPs?

A: Not at this time. We only support the PLP provided in the deployment platform packages.

Q: How can I query the PLP information?

A: Need to run xbutil query and look at the lower portion.

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# FPGA attestation for Azure NP-Series VMs (Preview)

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The FPGA Attestation service performs a series of validations on a design checkpoint file (called a "netlist") generated by the Xilinx toolset and produces a file that contains the validated image (called a "bitstream") that can be loaded onto the Xilinx U250 FPGA card in an NP series VM.

## News

The current attestation service is using Vitis 2020.2 from Xilinx, on Jan 17th 2022, we'll be moving to Vitis 2021.1, the change should be transparent to most users. Once your designs are "attested" using Vitis 2021.1, you should be moving to XRT2021.1. Xilinx will publish new marketplace images based on XRT 2021.1. Please note that current designs already attested on Vitis 2020.2, will work on the current deployment marketplace images as well as new images based on XRT2021.1.

As part of the move to 2021.1, Xilinx introduced a new DRC that might affect some designs previously working on Vitis 2020.2 regarding BUFCE\_LEAF failing attestation, for more details here: [Xilinx AR 75980 UltraScale/UltraScale+ BRAM: CLOCK\\_DOMAIN = Common Mode skew checks](#).

## Prerequisites

You will need an Azure subscription and an Azure Storage account. The subscription gives you access to Azure and the storage account is used to hold your netlist and output files of the attestation service.

We provide PowerShell and Bash scripts to submit attestation requests. The scripts use Azure CLI, which can run on Windows and Linux. PowerShell can run on Windows, Linux, and macOS.

[Azure CLI download \(required\)](#)

[PowerShell for Windows, Linux, and macOS download \(only for PowerShell scripts\)](#)

You will need to have your tenant and subscription ID authorized to submit to the attestation service. Visit <https://aka.ms/AzureFPGAAttestationPreview> to request access.

## Building your design for attestation

The preferred Xilinx toolset for building designs is Vitis 2020.2. Netlist files that were created with an earlier version of the toolset and are still compatible with 2020.2 can be used. Make sure you have loaded the correct shell to build against. The currently supported version is `xilinx_u250_gen3x16_xdma_2_1_202010_1`. The support files can be downloaded from the Xilinx Alveo lounge.

You must include the following argument to Vitis (v++ cmd line) to build an `xclbin` file that contains a netlist instead of a bitstream.

```
--advanced.param compiler.acceleratorBinaryContent=dcp
```

## Logging into Azure

Prior to performing any operations with Azure, you must log into Azure and set the subscription that is

authorized to call the service. Use the `az login` and `az account set -s <Sub ID or Name>` commands for this purpose. Further information about this process is documented here: [Sign in with Azure CLI](#). Use either the `sign in interactively` or `sign in with credentials` option on the command line.

## Creating a storage account and blob container

Your netlist file must be uploaded to an Azure storage blob container for access by the attestation service.

For more information on creating the account, a container, and uploading your netlist as a blob to that container, see [Quickstart: Create, download, and list blobs with Azure CLI](#).

You can also use the Azure portal for this as well.

## Upload your netlist file to Azure blob storage

There are several ways to copy the file; an example using the `az storage upload` cmdlet is shown below. The `az` commands run on both Linux and Windows. You can choose any name for the "blob" name but make sure to retain the `xclbin` extension.

```
az storage blob upload --account-name <storage account to receive netlist> --container-name <blob container name> --name <blob filename> --file <local file with netlist>
```

## Download the attestation scripts

The Validation scripts can be downloaded from the following Azure storage blob container:

<https://fpgaattestation.blob.core.windows.net/validationscripts/validate.zip>

The zip file has two PowerShell scripts, one to submit and the other to monitor while the third file is a bash script which performs both functions.

## Running the attestation scripts

To run the scripts, you will need to provide the name of your storage account, the name of the blob container where the netlist file is stored and the name of the netlist file. You will also need to create a Service shared access signature (SAS) that grants read/write access to your container (not the netlist). This SAS is used by the attestation service to make a local copy of your netlist file and to write back the resulting output files of the validation process to your container.

An overview of shared access signatures is available here with specific information about the Service SAS available here. The Service SAS page includes an important caution about protecting the generated SAS. Read the caution to understand the need to keep the SAS protected from malicious or unintended use.

You can generate a SAS for your container using the `az storage container generate-sas` cmdlet. Specify an expiry time in UTC format that is at least a few hours past the time of submission; around 6 hours should be more than adequate.

If you wish to use virtual directories, you must include the directory hierarchy as part of the container argument. For example, if you have a container named "netlists" and have a virtual directory named "image1" that contains the netlist blob, you would specify "netlists/image1" as the container name. Append any additional directory names to specify a deeper hierarchy.

### PowerShell

```
$sas=$(az storage container generate-sas --account-name <storage acct name> --name <blob container name> --https-only --permissions rwc --expiry <e.g., 2021-01-07T17:00Z> --output tsv)

.\Validate-FPGAIImage.ps1 -StorageAccountName <storage acct name> -Container <blob container name> -BlobContainerSAS $sas -NetlistName <netlist blob filename>
```

## Bash

```
sas=az storage container generate-sas --account-name <storage acct name> --name <blob container name> --https-only --permissions rwc --expiry <2021-01-07T17:00Z> --output tsv

validate-fpgaimage.sh --storage-account <storage acct name> --container <blob container name> --netlist-name <netlist blob filename> --blob-container-sas $sas
```

## Checking on the status of your submission

The Attestation service will return the orchestration ID of your submission. The submission scripts automatically start monitoring the submission by polling for completion. The orchestration ID is the primary way for us to review what happened to your submission so please keep that in case you have an issue. As reference points, attestation takes about 30 minutes to complete for a small netlist file (300MB in size); a 1.6GB file took an hour.

You can call the Monitor-Validation.ps1 script at any time to get status and results of attestation, providing the orchestration ID as an argument:

```
.\Monitor-Validation.ps1 -OrchestrationId <orchestration ID>
```

Alternatively, you can submit HTTP post request to the attestation service endpoint:

```
https://fpga-attestation.azurewebsites.net/api/ComputeFPGA_HttpGetStatus
```

The request body should contain your Subscription ID, Tenant ID, and orchestration ID of your attestation request:

```
{
  "OrchestrationId": "<orchestration ID>",
  "ClientSubscriptionId": "<your subscription ID>",
  "ClientTenantId": "<your tenant ID>"
}
```

## Post validation steps

The service will write its output back to your container. If the validation pass succeeds, your container will have the original netlist file (abc.xclbin), a file with the bitstream (abc.bit.xclbin), a file that identifies the private location of your stored bitstream (abc.azure.xclbin) and four log files: one for the startup process (abc-log.txt) and one each for the three parallel phases that perform the validation. These are named \*logPhaseX.txt where X is a number for the phase. The azure.xclbin is used on your VM to signal the uploading of your validated image to the U250.

If validation failed, an error-\*.txt file is written indicating which step failed. Also check the log files if the error log indicates that attestation failed. When contacting us for support, please be sure to include all these files as part of the support request along with the orchestration ID.

You can use the Azure portal to create your container as well as uploading your netlist and downloading the bitstream and log files. Submitting an attestation request and monitoring its progress through the portal is not supported at this time and must be done through scripts as described above.

# High performance computing VM sizes

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

Azure H-series virtual machines (VMs) are designed to deliver leadership-class performance, scalability, and cost efficiency for various real-world HPC workloads.

**HBv3-series** VMs are optimized for HPC applications such as fluid dynamics, explicit and implicit finite element analysis, weather modeling, seismic processing, reservoir simulation, and RTL simulation. HBv3 VMs feature up to 120 AMD EPYC™ 7003-series (Milan) CPU cores, 448 GB of RAM, and no hyperthreading. HBv3-series VMs also provide 350 GB/sec of memory bandwidth, up to 32 MB of L3 cache per core, up to 7 GB/s of block device SSD performance, and clock frequencies up to 3.5 GHz.

All HBv3-series VMs feature 200 Gb/sec HDR InfiniBand from NVIDIA Networking to enable supercomputer-scale MPI workloads. These VMs are connected in a non-blocking fat tree for optimized and consistent RDMA performance. The HDR InfiniBand fabric also supports Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is strongly recommended.

**HBv2-series** VMs are optimized for applications driven by memory bandwidth, such as fluid dynamics, finite element analysis, and reservoir simulation. HBv2 VMs feature 120 AMD EPYC 7742 processor cores, 4 GB of RAM per CPU core, and no simultaneous multithreading. Each HBv2 VM provides up to 340 GB/sec of memory bandwidth, and up to 4 teraFLOPS of FP64 compute.

HBv2 VMs feature 200 Gb/sec Mellanox HDR InfiniBand, while both HB and HC-series VMs feature 100 Gb/sec Mellanox EDR InfiniBand. Each of these VM types is connected in a non-blocking fat tree for optimized and consistent RDMA performance. HBv2 VMs support Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is strongly recommended.

**HB-series** VMs are optimized for applications driven by memory bandwidth, such as fluid dynamics, explicit finite element analysis, and weather modeling. HB VMs feature 60 AMD EPYC 7551 processor cores, 4 GB of RAM per CPU core, and no hyperthreading. The AMD EPYC platform provides more than 260 GB/sec of memory bandwidth.

**HC-series** VMs are optimized for applications driven by dense computation, such as implicit finite element analysis, molecular dynamics, and computational chemistry. HC VMs feature 44 Intel Xeon Platinum 8168 processor cores, 8 GB of RAM per CPU core, and no hyperthreading. The Intel Xeon Platinum platform supports Intel's rich ecosystem of software tools such as the Intel Math Kernel Library.

**H-series** VMs are optimized for applications driven by high CPU frequencies or large memory per core requirements. H-series VMs feature 8 or 16 Intel Xeon E5 2667 v3 processor cores, 7 or 14 GB of RAM per CPU core, and no hyperthreading. H-series features 56 Gb/sec Mellanox FDR InfiniBand in a non-blocking fat tree configuration for consistent RDMA performance. H-series VMs support Intel MPI 5.x and MS-MPI.

**NOTE**

All HBv3, HBv2, HB, and HC-series VMs have exclusive access to the physical servers. There is only 1 VM per physical server and there is no shared multi-tenancy with any other VMs for these VM sizes.

**NOTE**

The [A8 – A11 VMs](#) are retired as of 3/2021. No new VM deployments of these sizes are now possible. If you have existing VMs, refer to emailed notifications for next steps including migrating to other VM sizes in [HPC Migration Guide](#).

## RDMA-capable instances

Most of the HPC VM sizes feature a network interface for remote direct memory access (RDMA) connectivity. Selected [N-series](#) sizes designated with 'r' are also RDMA-capable. This interface is in addition to the standard Azure Ethernet network interface available in the other VM sizes.

This secondary interface allows the RDMA-capable instances to communicate over an InfiniBand (IB) network, operating at HDR rates for HBv3, HBv2, EDR rates for HB, HC, NDv2, and FDR rates for H16r, H16mr, and other RDMA-capable N-series virtual machines. These RDMA capabilities can boost the scalability and performance of Message Passing Interface (MPI) based applications.

**NOTE**

**SR-IOV support:** In Azure HPC, currently there are two classes of VMs depending on whether they are SR-IOV enabled for InfiniBand. Currently, almost all the newer generation, RDMA-capable or InfiniBand enabled VMs on Azure are SR-IOV enabled except for H16r, H16mr, and NC24r. RDMA is only enabled over the InfiniBand (IB) network and is supported for all RDMA-capable VMs. IP over IB is only supported on the SR-IOV enabled VMs. RDMA is not enabled over the Ethernet network.

- **Operating System** - Linux distributions such as CentOS, RHEL, Ubuntu, SUSE are commonly used. Windows Server 2016 and newer versions are supported on all the HPC series VMs. Windows Server 2012 R2 and Windows Server 2012 are also supported on the non-SR-IOV enabled VMs. Note that [Windows Server 2012 R2 is not supported on HBv2 onwards as VM sizes with more than 64 \(virtual or physical\) cores](#). See [VM Images](#) for a list of supported VM Images on the Marketplace and how they can be configured appropriately. The respective VM size pages also list out the software stack support.
- **InfiniBand and Drivers** - On InfiniBand enabled VMs, the appropriate drivers are required to enable RDMA. See [VM Images](#) for a list of supported VM Images on the Marketplace and how they can be configured appropriately. Also see [enabling InfiniBand](#) to learn about VM extensions or manual installation of InfiniBand drivers.
- **MPI** - The SR-IOV enabled VM sizes on Azure allow almost any flavor of MPI to be used with Mellanox OFED. On non-SR-IOV enabled VMs, supported MPI implementations use the Microsoft Network Direct (ND) interface to communicate between VMs. Hence, only Intel MPI 5.x and Microsoft MPI (MS-MPI) 2012 R2 or later versions are supported. Later versions of the Intel MPI runtime library may or may not be compatible with the Azure RDMA drivers. See [Setup MPI for HPC](#) for more details on setting up MPI on HPC VMs on Azure.

#### NOTE

**RDMA network address space:** The RDMA network in Azure reserves the address space 172.16.0.0/16. To run MPI applications on instances deployed in an Azure virtual network, make sure that the virtual network address space does not overlap the RDMA network.

## Cluster configuration options

Azure provides several options to create clusters of HPC VMs that can communicate using the RDMA network, including:

- **Virtual machines** - Deploy the RDMA-capable HPC VMs in the same scale set or availability set (when you use the Azure Resource Manager deployment model). If you use the classic deployment model, deploy the VMs in the same cloud service.
- **Virtual machine scale sets** - In a virtual machine scale set, ensure that you limit the deployment to a single placement group for InfiniBand communication within the scale set. For example, in a Resource Manager template, set the `singlePlacementGroup` property to `true`. Note that the maximum scale set size that can be spun up with `singlePlacementGroup=true` is capped at 100 VMs by default. If your HPC job scale needs are higher than 100 VMs in a single tenant, you may request an increase, [open an online customer support request](#) at no charge. The limit on the number of VMs in a single scale set can be increased to 300. Note that when deploying VMs using Availability Sets the maximum limit is at 200 VMs per Availability Set.

#### NOTE

**MPI among virtual machines:** If RDMA (e.g. using MPI communication) is required between virtual machines (VMs), ensure that the VMs are in the same virtual machine scale set or availability set.

- **Azure CycleCloud** - Create an HPC cluster using [Azure CycleCloud](#) to run MPI jobs.
- **Azure Batch** - Create an [Azure Batch](#) pool to run MPI workloads. To use compute-intensive instances when running MPI applications with Azure Batch, see [Use multi-instance tasks to run Message Passing Interface \(MPI\) applications in Azure Batch](#).
- **Microsoft HPC Pack** - [HPC Pack](#) includes a runtime environment for MS-MPI that uses the Azure RDMA network when deployed on RDMA-capable Linux VMs. For example deployments, see [Set up a Linux RDMA cluster with HPC Pack to run MPI applications](#).

## Deployment considerations

- **Azure subscription** – To deploy more than a few compute-intensive instances, consider a pay-as-you-go subscription or other purchase options. If you're using an [Azure free account](#), you can use only a limited number of Azure compute cores.
- **Pricing and availability** - Check [VM pricing](#) and [availability](#) by Azure regions.
- **Cores quota** – You might need to increase the cores quota in your Azure subscription from the default value. Your subscription might also limit the number of cores you can deploy in certain VM size families, including the H-series. To request a quota increase, [open an online customer support request](#) at no charge. (Default limits may vary depending on your subscription category.)

#### NOTE

Contact Azure Support if you have large-scale capacity needs. Azure quotas are credit limits, not capacity guarantees. Regardless of your quota, you are only charged for cores that you use.

- **Virtual network** – An Azure [virtual network](#) is not required to use the compute-intensive instances. However, for many deployments you need at least a cloud-based Azure virtual network, or a site-to-site connection if you need to access on-premises resources. When needed, create a new virtual network to deploy the instances. Adding compute-intensive VMs to a virtual network in an affinity group is not supported.
- **Resizing** – Because of their specialized hardware, you can only resize compute-intensive instances within the same size family (H-series or N-series). For example, you can only resize an H-series VM from one H-series size to another. Additional considerations around InfiniBand driver support and NVMe disks may need to be considered for certain VMs.

## Other sizes

- [General purpose](#)
- [Compute optimized](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [Previous generations](#)

## Next steps

- Learn more about [configuring your VMs, enabling InfiniBand, setting up MPI](#) and optimizing HPC applications for Azure at [HPC Workloads](#).
- Review the [HBv3-series overview](#) and [HC-series overview](#).
- Read about the latest announcements, HPC workload examples, and performance results at the [Azure Compute Tech Community Blogs](#).
- For a higher level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).

# H-series

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

H-series VMs are optimized for applications driven by high CPU frequencies or large memory per core requirements. H-series VMs feature 8 or 16 Intel Xeon E5 2667 v3 processor cores, up to 14 GB of RAM per CPU core, and no hyperthreading. H-series features 56 Gb/sec Mellanox FDR InfiniBand in a non-blocking fat tree configuration for consistent RDMA performance. H-series VMs are not SR-IOV enabled currently and support Intel MPI 5.x and MS-MPI.

**ACU:** 290-300

**Premium Storage:** Not Supported

**Premium Storage caching:** Not Supported

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1

**Accelerated Networking:** Not Supported

**Ephemeral OS Disks:** Not Supported

SIZE	VCP U	PRO CES SOR	ME MO RY (GIB )	ME MO RY BAN DWI DTH GB/ S	BAS E CPU FRE QUE NCY (GH Z)	ALL- COR ES FRE QUE NCY (GH Z, PEA K)	SIN GLE- COR E FRE QUE NCY (GH Z, PEA K)	RDM A PER FOR MA NCE (GB/ S)	TEM P STO RAG E (GIB )	MAX DAT A DISK S	MAX DISK THR oug HPU T: IOPS	MAX ETH ERN ET VNI CS	
Standard _H8	8	Intel Xeon E5 2667 v3	56	40	3.2	3.3	3.6	-	Intel 5.x, MS-MPI	1000	32	32 x 500	2
Standard _H16	16	Intel Xeon E5 2667 v3	112	80	3.2	3.3	3.6	-	Intel 5.x, MS-MPI	2000	64	64 x 500	4
Standard _H8m	8	Intel Xeon E5 2667 v3	112	40	3.2	3.3	3.6	-	Intel 5.x, MS-MPI	1000	32	32 x 500	2
Standard _H16m	16	Intel Xeon E5 2667 v3	224	80	3.2	3.3	3.6	-	Intel 5.x, MS-MPI	2000	64	64 x 500	4

SIZE	VCP U	PRO CES SOR	ME MO RY (GIB )	ME MO RY BAN DWI DTH GB/ S	BAS E CPU FRE QUE NCY (GH Z, PEA K)	ALL- COR ES FRE QUE NCY (GH Z, PEA K)	SIN GLE- COR E FRE QUE NCY (GH Z, PEA K)	RDM A PER FOR MA NCE (GB/ S)	TEM P STO RAG E SUP POR T	MAX DISK THR OUG HPU T: IOPS	MAX ETH ERN ET VNI CS		
Standard _H1 6r <sup>1</sup>	16	Intel Xeo n E5 266 7 v3	112	80	3.2	3.3	3.6	56	Intel 5.x, MS- MPI	200 0	64	64 x 500	4
Standard _H1 6mr 1	16	Intel Xeo n E5 266 7 v3	224	80	3.2	3.3	3.6	56	Intel 5.x, MS- MPI	200 0	64	64 x 500	4

<sup>1</sup> For MPI applications, dedicated RDMA backend network is enabled by FDR InfiniBand network.

#### NOTE

Among the [RDMA capable VMs](#), the H-series are not SR-IOV enabled. Therefore, the supported [VM Images](#), [InfiniBand driver](#) requirements and supported [MPI libraries](#) are different from the SR-IOV enabled VMs.

A quirk of the alternate NIC virtualization solution in place for the H-series is that the OS may occasionally report inaccurate link speeds for the synthetic NIC that is used for RDMA connections. This issue does not, however, impact actual performance experienced by jobs using the VM's RDMA capability, so outputs like the following are not a cause for concern.

```
$ ethtool eth1
Settings for eth1:
...
Speed: 10000Mb/s
```

## Software specifications

SOFTWARE SPECIFICATIONS	H-SERIES VM
Max MPI Job Size	4800 cores (300 VMs in a single virtual machine scale set with singlePlacementGroup=true)
MPI Support	Intel MPI 5.x, MS-MPI
OS Support for non-SRIOV RDMA	CentOS/RHEL 6.5 - 7.4, SLES 12 SP4+, WinServer 2012 - 2016
Orchestrator Support	CycleCloud, Batch, AKS

## Get Started

- [Overview](#) of HPC on InfiniBand-enabled H-series and N-series VMs.

- [Configuring VMs](#) and supported [OS and VM Images](#).
- [Enabling InfiniBand](#) with HPC VM images, VM extensions or manual installation.
- [Setting up MPI](#), including code snippets and recommendations.
- [Cluster configuration options](#).
- [Deployment considerations](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

- Read about the latest announcements, HPC workload examples, and performance results at the [Azure Compute Tech Community Blogs](#).
- For a higher level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).
- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# HB-series

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

HB-series VMs are optimized for applications that are driven by memory bandwidth, such as fluid dynamics, explicit finite element analysis, and weather modeling. HB VMs feature 60 AMD EPYC 7551 processor cores, 4 GB of RAM per CPU core, and no simultaneous multithreading. An HB VM provides up to 260 GB/sec of memory bandwidth.

HB-series VMs feature 100 Gb/sec Mellanox EDR InfiniBand. These VMs are connected in a non-blocking fat tree for optimized and consistent RDMA performance. These VMs support Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is recommended.

**ACU:** 199-216

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported ([Learn more](#) about performance and potential issues)

**Ephemeral OS Disks:** Supported

SIZE	VCP U	PRO CESS OR	MEM ORY (GB)	MEM ORY BAN DTH GB/S	BASE CPU FREQ	ALL-CORE S FREQ UEN CY (GHZ , PEAK )	SING LE-CORE FREQ UEN CY (GHZ , PEAK )	RDM A PERFOR MANCE (GB/S)	MPI SUPP ORT	TEM P STOR AGE (GB)	MAX DATA DISK S	MAX ETHERNET VNIC S
Standard _HB60rs	60	AMD EPYC 7551	228	263	2.0	2.55	2.55	100	All	700	4	8

Learn more about the:

- [Architecture and VM topology](#)
- Supported [software stack](#) including supported OS
- Expected [performance](#) of the HB-series VM

## Get Started

- [Overview](#) of HPC on InfiniBand-enabled H-series and N-series VMs.
- [Configuring VMs](#) and supported [OS and VM Images](#).
- [Enabling InfiniBand](#) with HPC VM images, VM extensions or manual installation.
- [Setting up MPI](#), including code snippets and recommendations.

- [Cluster configuration options](#).
- [Deployment considerations](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

- Read about the latest announcements, HPC workload examples, and performance results at the [Azure Compute Tech Community Blogs](#).
- For a higher level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).
- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# HBv2-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

HBv2-series VMs are optimized for applications that are driven by memory bandwidth, such as fluid dynamics, finite element analysis, and reservoir simulation. HBv2 VMs feature 120 AMD EPYC 7742 processor cores, 4 GB of RAM per CPU core, and no simultaneous multithreading. Each HBv2 VM provides up to 340 GB/sec of memory bandwidth, and up to 4 teraFLOPS of FP64 compute.

HBv2-series VMs feature 200 Gb/sec Mellanox HDR InfiniBand. These VMs are connected in a non-blocking fat tree for optimized and consistent RDMA performance. These VMs support Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is recommended.

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported ([Learn more](#) about performance and potential issues)

**Ephemeral OS Disks:** Supported

SIZE	VCP U	PRO CESS OR	MEM ORY (GIB)	MEM BAN DWI DTH GB/S	BASE CPU FREQ UEN CY (GHZ )	ALL-CORE S FREQ UEN CY (GHZ , PEAK )	SING LE-CORE FREQ UEN CY (GHZ , PEAK )	RDM A PERFOR MANCE (GB/S)	MPI SUPP ORT	TEM P STOR AGE (GIB)	MAX DATA DISK S	MAX ETHERNET VNIC S
Standard _HB1 20rs_v2	120	AMD EPYC 7V12	456	350	2.45	3.1	3.3	200	All	480 + 960	8	8

Learn more about the:

- [Architecture and VM topology](#)
- Supported [software stack](#) including supported OS
- Expected [performance](#) of the HBv2-series VM

## Get Started

- [Overview](#) of HPC on InfiniBand-enabled H-series and N-series VMs.
- Configuring VMs and supported [OS and VM Images](#).
- Enabling [InfiniBand](#) with HPC VM images, VM extensions or manual installation.
- Setting up [MPI](#), including code snippets and recommendations.

- [Cluster configuration options](#).
- [Deployment considerations](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

- Read about the latest announcements, HPC workload examples, and performance results at the [Azure Compute Tech Community Blogs](#).
- For a higher level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).
- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# HBv3-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

HBv3-series VMs are optimized for HPC applications such as fluid dynamics, explicit and implicit finite element analysis, weather modeling, seismic processing, reservoir simulation, and RTL simulation. HBv3 VMs feature up to 120 AMD EPYC™ 7V73X (Milan-X) CPU cores, 448 GB of RAM, and no hyperthreading. HBv3-series VMs also provide 350 GB/sec of memory bandwidth (amplified up to 630 GB/s), up to 96 MB of L3 cache per core (1.536 GB total per VM), up to 7 GB/s of block device SSD performance, and clock frequencies up to 3.5 GHz.

All HBv3-series VMs feature 200 Gb/sec HDR InfiniBand from NVIDIA Networking to enable supercomputer-scale MPI workloads. These VMs are connected in a non-blocking fat tree for optimized and consistent RDMA performance. The HDR InfiniBand fabric also supports Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is strongly recommended.

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported ([Learn more](#) about performance and potential issues)

**Ephemeral OS Disks:** Supported

SIZE	VCP U	PRO CESS OR	MEM ORY (GIB)	MEM BAN DWI DTH GB/S	BASE CPU FREQ UEN CY (GHZ )	ALL-CORE S FREQ UEN CY (GHZ , PEAK )	SING LE-CORE FREQ UEN CY (GHZ , PEAK )	RDM A PERFOR MANCE (GB/S)	MPI SUPP ORT	TEM P STOR AGE (GIB)	MAX DATA DISK S	MAX ETHERNET VNIC S
Standard _HB1 20rs_ v3	120	AMD EPYC 7V73 X	448	350	1.9	3.0	3.5	200	All	2 * 960	32	8
Standard _HB1 20- 96rs_ v3	96	AMD EPYC 7V73 X	448	350	1.9	3.0	3.5	200	All	2 * 960	32	8

SIZE	VCP U	PRO CESS OR	MEM ORY (GB)	MEM ORY BAN DWI DTH GB/S	BASE CPU FREQ UE CY (GHZ )	ALL- CORE S FREQ UE CY (GHZ , PEAK )	SING LE- CORE FREQ UE CY (GHZ , PEAK )	RDM A PERF ORM ANC E (GB/ S)	MPI SUPP ORT	TEM P STOR AGE (GiB)	MAX DATA DISK S	MAX ETHE RNET VNIC S
Standard_HB1_20-64rs_v3	64	AMD EPYC 7V73X	448	350	1.9	3.0	3.5	200	All	2 * 960	32	8
Standard_HB1_20-32rs_v3	32	AMD EPYC 7V73X	448	350	1.9	3.0	3.5	200	All	2 * 960	32	8
Standard_HB1_20-16rs_v3	16	AMD EPYC 7V73X	448	350	1.9	3.0	3.5	200	All	2 * 960	32	8

Learn more about the:

- [Architecture and VM topology](#)
- Supported [software stack](#) including supported OS
- Expected [performance](#) of the HBv3-series VM

## Get Started

- [Overview](#) of HPC on InfiniBand-enabled H-series and N-series VMs.
- [Configuring VMs](#) and supported [OS and VM Images](#).
- [Enabling InfiniBand](#) with HPC VM images, VM extensions or manual installation.
- [Setting up MPI](#), including code snippets and recommendations.
- [Cluster configuration options](#).
- [Deployment considerations](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.

- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator : [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

- Read about the latest announcements, HPC workload examples, and performance results at the [Azure Compute Tech Community Blogs](#).
- For a higher level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).
- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# HC-series

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

HC-series VMs are optimized for applications driven by dense computation, such as implicit finite element analysis, molecular dynamics, and computational chemistry. HC VMs feature 44 Intel Xeon Platinum 8168 processor cores, 8 GB of RAM per CPU core, and no hyperthreading. The Intel Xeon Platinum platform supports Intel's rich ecosystem of software tools such as the Intel Math Kernel Library and advanced vector processing capabilities such as AVX-512.

HC-series VMs feature 100 Gb/sec Mellanox EDR InfiniBand. These VMs are connected in a non-blocking fat tree for optimized and consistent RDMA performance. These VMs support Adaptive Routing and the Dynamic Connected Transport (DCT, in addition to standard RC and UD transports). These features enhance application performance, scalability, and consistency, and their usage is recommended.

**ACU:** 297-315

**Premium Storage:** Supported

**Premium Storage caching:** Supported

**Ultra Disks:** Supported ([Learn more](#) about availability, usage and performance)

**Live Migration:** Not Supported

**Memory Preserving Updates:** Not Supported

**VM Generation Support:** Generation 1 and 2

**Accelerated Networking:** Supported ([Learn more](#) about performance and potential issues)

**Ephemeral OS Disks:** Supported

SIZE	VCP U	PRO CESS OR	MEM ORY (GIB)	MEM BAN DWI DTH GB/S	BASE CPU FREQ UEN CY (GHZ )	ALL-CORE S FREQ UEN CY (GHZ , PEAK )	SING LE-CORE FREQ UEN CY (GHZ , PEAK )	RDM A PERFOR MANCE (GB/S)	MPI SUPP ORT	TEM P STOR AGE (GIB)	MAX DATA DISK S	MAX ETHERNET VNIC S
Standard _HC4 4rs	44	Intel Xeon Platinum 8168	352	191	2.7	3.4	3.7	100	All	700	4	8

Learn more about the:

- [Architecture and VM topology](#)
- Supported [software stack](#) including supported OS
- Expected [performance](#) of the HC-series VM

## Get Started

- [Overview](#) of HPC on InfiniBand-enabled H-series and N-series VMs.
- [Configuring VMs](#) and supported [OS and VM Images](#).

- [Enabling InfiniBand](#) with HPC VM images, VM extensions or manual installation.
- [Setting up MPI](#), including code snippets and recommendations.
- [Cluster configuration options](#).
- [Deployment considerations](#).

## Size table definitions

- Storage capacity is shown in units of GiB or  $1024^3$  bytes. When you compare disks measured in GB ( $1000^3$  bytes) to disks measured in GiB ( $1024^3$ ) remember that capacity numbers given in GiB may appear smaller. For example, 1023 GiB = 1098.4 GB.
- Disk throughput is measured in input/output operations per second (IOPS) and MBps where MBps =  $10^6$  bytes/sec.
- Data disks can operate in cached or uncached modes. For cached data disk operation, the host cache mode is set to **ReadOnly** or **ReadWrite**. For uncached data disk operation, the host cache mode is set to **None**.
- To learn how to get the best storage performance for your VMs, see [Virtual machine and disk performance](#).
- **Expected network bandwidth** is the maximum aggregated bandwidth allocated per VM type across all NICs, for all destinations. For more information, see [Virtual machine network bandwidth](#).

Upper limits aren't guaranteed. Limits offer guidance for selecting the right VM type for the intended application. Actual network performance will depend on several factors including network congestion, application loads, and network settings. For information on optimizing network throughput, see [Optimize network throughput for Azure virtual machines](#). To achieve the expected network performance on Linux or Windows, you may need to select a specific version or optimize your VM. For more information, see [Bandwidth/Throughput testing \(NTTCP\)](#).

## Other sizes and information

- [General purpose](#)
- [Memory optimized](#)
- [Storage optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Previous generations](#)

Pricing Calculator: [Pricing Calculator](#)

For more information on disk types, see [What disk types are available in Azure?](#)

## Next steps

- Read about the latest announcements, HPC workload examples and performance results at the [Azure Compute Tech Community Blogs](#).
- For a high-level architectural view of running HPC workloads, see [High Performance Computing \(HPC\) on Azure](#).
- Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Migrate your H and H-series Promo virtual machines by August 31, 2022

9/21/2022 • 3 minutes to read • [Edit Online](#)

Microsoft Azure has introduced newer generations of high-performance computing (HPC), general purpose, and memory-optimized virtual machines (VMs). For this reason, we recommend that you migrate workloads from the original H-series and H-series Promo VMs to our newer offerings.

Azure [HC](#), [HBv2](#), [HBv3](#), [Dv4](#), [Dav4](#), [Ev4](#), and [Eav4](#) VMs have greater memory bandwidth, improved networking capabilities, and better cost and performance across various HPC workloads. On August 31, 2022, we're retiring the following H-series Azure VM sizes:

- H8
- H8m
- H16
- H16r
- H16m
- H16mr
- H8 Promo
- H8m Promo
- H16 Promo
- H16r Promo
- H16m Promo
- H16mr Promo

## How does the H-series migration affect me?

1-year and 3-year RI offerings for the VMs are no longer available; however, regular PAYGO offer can still be transacted until the official decommission date. After August 31, 2022, any remaining H-series VM subscriptions in the preceding list will be set to a deallocated state. They'll stop working and no longer incur billing charges. If you need RI, please refer to our migration documents to find the suitable VM offerings that have RI available.

You can either exchange/refund your existing reservations. If you choose not to, after the hardware is deprecated you won't be getting the reservation benefit but still be paying for it.

The current VM size retirement only affects the VM sizes in the [H series](#), which includes the H-series Promo.

## What actions should I take?

You'll need to resize or deallocate your H-series VMs. We recommend that you migrate workloads from the original H-series VMs and the H-series Promo VMs to our newer offerings.

**HPC workloads:** [HC](#), [HBv2](#), and [HBv3](#) VMs offer substantially higher levels of HPC workload performance and cost efficiency because of:

- Large improvements in CPU core architecture.
- Higher memory bandwidth.
- Larger L3 caches.
- Enhanced InfiniBand networking hardware and software support as compared to H series.

As a result, HC, HBv2, and HBv3 series will in general offer substantially better performance per unit of cost (maximizing performance for a fixed amount of spend) and cost per performance (minimizing cost for a fixed amount of performance).

**General purpose workloads:** [Dv4](#), [Dav4](#), and Dv5 VMs offer the same or better CPU performance at identical or larger core counts, a comparable amount of memory per physical CPU core, better Azure networking capabilities, and lower overall cost.

**Memory-optimized workloads:** [Ev4](#), [Eav4](#), and Ev5 VMs offer the same or better CPU performance at identical or larger core counts, a comparable amount of memory per physical CPU core, better Azure networking capabilities, and lower overall cost.

[H-series](#) and H-series Promo VMs won't be retired until September 2022. We're providing this guide in advance to give you a long window to assess, plan, and execute your migration.

## Migration steps

1. Choose a series and size for migration.
2. Get a quota for the target VM series.
3. Resize the current H-series VM size to the target size.

## Breaking changes

If you use H-series VM sizes that expose an InfiniBand networking interface, such as those sizes with an "r" in the VM size name, and you want your new VM sizes to also support InfiniBand networking, you'll no longer be able to use legacy OS images with built-in InfiniBand driver support (CentOS 7.4 and prior, Windows Server 2012).

Instead, use modern OS images such as those available in Azure Marketplace that support modern operating systems (CentOS 7.5 and newer, Windows Server 2016 and newer) and standard OFED drivers. See the [supported software stack](#), which includes the supported OS for the respective VM sizes.

## Get a quota for the target VM family

Follow the guide to [request an increase in vCPU quota by VM family](#).

## Resize the current VM

You can [resize the virtual machine](#).

# Migrate your HB-series virtual machines by August 31, 2024

9/21/2022 • 2 minutes to read • [Edit Online](#)

Microsoft Azure has introduced HBv2 and HBv3-series virtual machines (VMs) for high-performance computing (HPC). For this reason, we recommend that you migrate workloads from original HB-series VMs to our newer offerings.

Azure [HBv2](#) and [HBv3](#) VMs have greater memory bandwidth, improved remote direct memory access (RDMA) networking capabilities, larger and faster local solid-state drives, and better cost and performance across various HPC workloads. As a result, we're retiring our HB-series Azure VM sizes on August 31, 2024.

## How does the HB-series migration affect me?

After August 31, 2024, any remaining HB-size VM subscriptions will be set to a deallocated state. They'll stop working and no longer incur billing charges.

### NOTE

This VM size retirement only affects the VM sizes in the HB series. This retirement announcement doesn't apply to the newer HBv2, HBv3, and HC-series VMs.

## What actions should I take?

You'll need to resize or deallocate your H-series VMs. We recommend that you migrate workloads from the original H-series VMs and the H-series Promo VMs to our newer offerings.

[HBv2](#) and [HBv3](#) VMs offer substantially higher levels of HPC workload performance and cost efficiency because of:

- Large improvements in CPU core architecture.
- Higher memory bandwidth.
- Larger L3 caches.
- Enhanced InfiniBand networking as compared to HB series.

As a result, HBv2 and HBv3 series will in general offer substantially better performance per unit of cost (maximizing performance for a fixed amount of spend) and cost per performance (minimizing cost for a fixed amount of performance).

All regions that contain HB-series VMs contain HBv2 and HBv3-series VMs. Existing workloads that run on HB-series VMs can be migrated without concern for geographic placement or for access to more services in those regions.

[HB-series](#) VMs won't be retired until September 2024. We're providing this guide in advance to give you a long window to assess, plan, and execute your migration.

### Recommendations for workload migration from HB-series VMs

CURRENT VM SIZE	TARGET VM SIZE	DIFFERENCE IN SPECIFICATION
Standard_HB60rs	Standard_HB120rs_v2 Standard_HB120rs_v3 Standard_HB120-64rs_v3	Newer CPU: AMD Rome and Milan (+20-30% IPC) Memory: Up to 2x more RAM Memory bandwidth: Up to 30% more memory bandwidth InfiniBand: 200 Gb HDR (2x higher bandwidth) Max data disks: Up to 32 (+8x)
Standard_HB60-45rs	Standard_HB120-96rs_v3 Standard_HB120-64rs_v3 Standard_HB120-32rs_v3	Newer CPU: AMD Rome and Milan (+20-30% IPC) Memory: Up to 2x more RAM Memory bandwidth: Up to 30% more memory bandwidth InfiniBand: 200 Gb HDR (2x higher bandwidth) Max data disks: Up to 32 (+8x)
Standard_HB60-30rs	Standard_HB120-32rs_v3 Standard_HB120-16rs_v3	Newer CPU: AMD Rome and Milan (+20-30% IPC) Memory: Up to 2x more RAM Memory bandwidth: Up to 30% more memory bandwidth InfiniBand: 200 Gb HDR (2x higher bandwidth) Max data disks: Up to 32 (+8x)
Standard_HB60-15rs	Standard_HB120-16rs_v3	Newer CPU: AMD Rome and Milan (+20-30% IPC) Memory: Up to 2x more RAM Memory bandwidth: Up to 30% more memory bandwidth InfiniBand: 200 Gb HDR (2x higher bandwidth) Max data disks: Up to 32 (+8x)

## Migration steps

1. Choose a series and size for migration.
2. Get a quota for the target VM series.
3. Resize the current HB-series VM size to the target size.

## Get a quota for the target VM family

Follow the guide to [request an increase in vCPU quota by VM family](#).

## Resize the current VM

You can [resize the virtual machine](#).

# Previous generations of virtual machine sizes

9/21/2022 • 15 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

## TIP

Try the [Virtual machines selector tool](#) to find other sizes that best fit your workload.

This section provides information on previous generations of virtual machine sizes. These sizes can still be used, but there are newer generations available.

## F-series

F-series is based on the 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) processor, which can achieve clock speeds as high as 3.1 GHz with the Intel Turbo Boost Technology 2.0. This is the same CPU performance as the Dv2-series of VMs.

F-series VMs are an excellent choice for workloads that demand faster CPUs but do not need as much memory or temporary storage per vCPU. Workloads such as analytics, gaming servers, web servers, and batch processing will benefit from the value of the F-series.

ACU: 210 - 250

Premium Storage: Not Supported

Premium Storage caching: Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THROUGHPUT: IOPS	MAX NICS/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_F1	1	2	16	3000/46/23	4/4x500	2/750
Standard_F2	2	4	32	6000/93/46	8/8x500	2/1500
Standard_F4	4	8	64	12000/187/93	16/16x500	4/3000
Standard_F8	8	16	128	24000/375/187	32/32x500	8/6000
Standard_F16	16	32	256	48000/750/375	64/64x500	8/12000

## Fs-series <sup>1</sup>

The Fs-series provides all the advantages of the F-series, in addition to Premium storage.

ACU: 210 - 250

Premium Storage: Supported

Premium Storage caching: Supported

Ephemeral OS Disks: Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS/MBPS (CACHE SIZE IN GiB)	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_F1s	1	2	4	4	4000/32 (12)	3200/48	2/750
Standard_F2s	2	4	8	8	8000/64 (24)	6400/96	2/1500
Standard_F4s	4	8	16	16	16000/128 (48)	12800/192	4/3000
Standard_F8s	8	16	32	32	32000/256 (96)	25600/384	8/6000
Standard_F16s	16	32	64	64	64000/512 (192)	51200/768	8/12000

MBps =  $10^6$  bytes per second, and GiB =  $1024^3$  bytes.

<sup>1</sup> The maximum disk throughput (IOPS or MBps) possible with a Fs series VM may be limited by the number, size, and striping of the attached disk(s). For details, see [Design for high performance](#).

## NVv2-series

Newer size recommendation: [NVv3-series](#)

The NVv2-series virtual machines are powered by [NVIDIA Tesla M60](#) GPUs and NVIDIA GRID technology with Intel Broadwell CPUs. These virtual machines are targeted for GPU accelerated graphics applications and virtual desktops where customers want to visualize their data, simulate results to view, work on CAD, or render and stream content. Additionally, these virtual machines can run single precision workloads such as encoding and rendering. NVv2 virtual machines support Premium Storage and come with twice the system memory (RAM) when compared with its predecessor NV-series.

Each GPU in NVv2 instances comes with a GRID license. This license gives you the flexibility to use an NV instance as a virtual workstation for a single user, or 25 concurrent users can connect to the VM for a virtual application scenario.

Ephemeral OS Disks: Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICS	VIRTUAL WORKSTATIONS	VIRTUAL APPLICATIONS
Standard_NV6s_v2	6	112	320	1	8	12	4	1	25
Standard_NV12s_v2	12	224	640	2	16	24	8	2	50
Standard_NV24s_v2	24	448	1280	4	32	32	8	4	100

## Older generations of virtual machine sizes

This section provides information on older generations of virtual machine sizes. These sizes are still supported but will not receive additional capacity. There are newer or alternative sizes that are generally available. Please refer to [Sizes virtual machines in Azure](#) to choose the VM sizes that will best fit your need.

For more information on resizing a Linux VM, see [Resize a VM](#).

### Basic A

Newer size recommendation: [Av2-series](#)

Premium Storage: Not Supported

Premium Storage caching: Not Supported

The basic tier sizes are primarily for development workloads and other applications that don't require load balancing, auto-scaling, or memory-intensive virtual machines.

SIZE - SIZE\NAME	VCPU	MEMORY	NICS (MAX)	MAX TEMPORARY DISK SIZE	MAX. DATA DISKS (1023 GB EACH)	MAX. IOPS (300 PER DISK)
A0\Basic_A0	1	768 MB	2	20 GB	1	1x300
A1\Basic_A1	1	1.75 GB	2	40 GB	2	2x300
A2\Basic_A2	2	3.5 GB	2	60 GB	4	4x300
A3\Basic_A3	4	7 GB	2	120 GB	8	8x300
A4\Basic_A4	8	14 GB	2	240 GB	16	16x300

### Standard A0 - A4 using CLI and PowerShell

In the classic deployment model, some VM size names are slightly different in CLI and PowerShell:

- Standard\_A0 is ExtraSmall

- Standard\_A1 is Small
- Standard\_A2 is Medium
- Standard\_A3 is Large
- Standard\_A4 is ExtraLarge

## A-series

Newer size recommendation: [Av2-series](#)

ACU: 50-100

Premium Storage: Not Supported

Premium Storage caching: Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (HDD): GIB	MAX DATA DISKS	MAX DATA DISK THROUGHPUT: IOPS	MAX NICS/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_A0 <sup>1</sup>	1	0.768	20	1	1x500	2/100
Standard_A1	1	1.75	70	2	2x500	2/500
Standard_A2	2	3.5	135	4	4x500	2/500
Standard_A3	4	7	285	8	8x500	2/1000
Standard_A4	8	14	605	16	16x500	4/2000
Standard_A5	2	14	135	4	4x500	2/500
Standard_A6	4	28	285	8	8x500	2/1000
Standard_A7	8	56	605	16	16x500	4/2000

<sup>1</sup> The A0 size is over-subscribed on the physical hardware. For this specific size only, other customer deployments may impact the performance of your running workload. The relative performance is outlined below as the expected baseline, subject to an approximate variability of 15 percent.

## A-series - compute-intensive instances

Newer size recommendation: [Av2-series](#)

ACU: 225

Premium Storage: Not Supported

Premium Storage caching: Not Supported

The A8-A11 and H-series sizes are also known as *compute-intensive instances*. The hardware that runs these sizes is designed and optimized for compute-intensive and network-intensive applications, including high-performance computing (HPC) cluster applications, modeling, and simulations. The A8-A11 series uses Intel Xeon E5-2670 @ 2.6 GHZ and the H-series uses Intel Xeon E5-2667 v3 @ 3.2 GHz.

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (HDD): GIB	MAX DATA DISKS	MAX DATA DISK THROUGHPUT: IOPS	MAX NICs
Standard_A8 <sup>1</sup>	8	56	382	32	32x500	2
Standard_A9 <sup>1</sup>	16	112	382	64	64x500	4
Standard_A10	8	56	382	32	32x500	2
Standard_A11	16	112	382	64	64x500	4

<sup>1</sup> For MPI applications, dedicated RDMA backend network is enabled by FDR InfiniBand network, which delivers ultra-low-latency and high bandwidth.

#### NOTE

The A8 – A11 VMs are planned for retirement on 3/2021. We strongly recommend not creating any new A8 – A11 VMs. Please migrate any existing A8 – A11 VMs to newer and powerful high-performance computing VM sizes such as H, HB, HC, HBv2 as well as general purpose compute VM sizes such as D, E, and F for better price-performance. For more information, see [HPC Migration Guide](#).

## D-series

Newer size recommendation: [Dav4-series](#), [Dv4-series](#) and [Ddv4-series](#)

ACU: 160-250<sup>1</sup>

Premium Storage: Not Supported

Premium Storage caching: Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THROUGHPUT: IOPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D1	1	3.5	50	3000/46/23	4/4x500	2/500
Standard_D2	2	7	100	6000/93/46	8/8x500	2/1000
Standard_D3	4	14	200	12000/187/93	16/16x500	4/2000
Standard_D4	8	28	400	24000/375/187	32/32x500	8/4000

<sup>1</sup> VM Family can run on one of the following CPU's: 2.2 GHz Intel Xeon® E5-2660 v2, 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) or 2.3 GHz Intel XEON® E5-2673 v4 (Broadwell)

## D-series - memory optimized

Newer size recommendation: [Dav4-series](#), [Dv4-series](#) and [Ddv4-series](#)

ACU: 160-250<sup>1</sup>

Premium Storage: Not Supported

Premium Storage caching: Not Supported

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THROUGHPUT: IOPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_D11	2	14	100	6000/93/46	8/8x500	2/1000
Standard_D12	4	28	200	12000/187/93	16/16x500	4/2000
Standard_D13	8	56	400	24000/375/187	32/32x500	8/4000
Standard_D14	16	112	800	48000/750/375	64/64x500	8/8000

<sup>1</sup> VM Family can run on one of the following CPU's: 2.2 GHz Intel Xeon® E5-2660 v2, 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) or 2.3 GHz Intel XEON® E5-2673 v4 (Broadwell)

## Preview: DC-series

Newer size recommendation: [DCsv2-series](#)

Premium Storage: Supported

Premium Storage caching: Supported

Ephemeral OS Disks: Supported

The DC-series uses the latest generation of 3.7GHz Intel XEON E-2176G Processor with SGX technology, and with the Intel Turbo Boost Technology can go up to 4.7GHz.

SIZE	VCPU	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS / MBPS (CACHE SIZE IN GIB)	MAX UNCACHED DISK THROUGHPUT: IOPS / MBPS	MAX NICs / EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_DC2s	2	8	100	2	4000 / 32 (43)	3200 / 48	2 / 1500
Standard_DC4s	4	16	200	4	8000 / 64 (86)	6400 / 96	2 / 3000

**IMPORTANT**

DC-series VMs are [generation 2 VMs](#) and only support [Gen2](#) images.

**DS-series**

Newer size recommendation: [Dasv4-series](#), [Dsv4-series](#) and [Ddsv4-series](#)

ACU: 160-250 <sup>1</sup>

Premium Storage: Supported

Premium Storage caching: Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS/MBPS (CACHE SIZE IN GIB)	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs/EXPE CTED NETWORK BANDWIDT H (MBPS)
Standard_DS1	1	3.5	7	4	4000/32 (43)	3200/32	2/500
Standard_DS2	2	7	14	8	8000/64 (86)	6400/64	2/1000
Standard_DS3	4	14	28	16	16000/128 (172)	12800/128	4/2000
Standard_DS4	8	28	56	32	32000/256 (344)	25600/256	8/4000

<sup>1</sup> VM Family can run on one of the following CPU's: 2.2 GHz Intel Xeon® E5-2660 v2, 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) or 2.3 GHz Intel XEON® E5-2673 v4 (Broadwell)

**DS-series - memory optimized**

Newer size recommendation: [Dasv4-series](#), [Dsv4-series](#) and [Ddsv4-series](#)

ACU: 160-250 <sup>1,2</sup>

Premium Storage: Supported

Premium Storage caching: Supported

SIZE	VCPUs	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS/MBPS (CACHE SIZE IN GiB)	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_D S11	2	14	28	8	8000/64 (72)	6400/64	2/1000
Standard_D S12	4	28	56	16	16000/128 (144)	12800/128	4/2000
Standard_D S13	8	56	112	32	32000/256 (288)	25600/256	8/4000
Standard_D S14	16	112	224	64	64000/512 (576)	51200/512	8/8000

<sup>1</sup> The maximum disk throughput (IOPS or MBps) possible with a DS series VM may be limited by the number, size and striping of the attached disk(s). For details, see [Design for high performance](#). <sup>2</sup> VM Family can run on one of the following CPU's: 2.2 GHz Intel Xeon® E5-2660 v2, 2.4 GHz Intel Xeon® E5-2673 v3 (Haswell) or 2.3 GHz Intel XEON® E5-2673 v4 (Broadwell)

## Ls-series

Newer size recommendation: [Lsv2-series](#)

The Ls-series offers up to 32 vCPUs, using the Intel® Xeon® processor E5 v3 family. The Ls-series gets the same CPU performance as the G/GS-Series and comes with 8 GiB of memory per vCPU.

The Ls-series does not support the creation of a local cache to increase the IOPS achievable by durable data disks. The high throughput and IOPS of the local disk makes Ls-series VMs ideal for NoSQL stores such as Apache Cassandra and MongoDB which replicate data across multiple VMs to achieve persistence in the event of the failure of a single VM.

ACU: 180-240

Premium Storage: Supported

Premium Storage caching: Not Supported

Ephemeral OS Disks: Supported

SIZE	VCPUs	MEMORY (GiB)	TEMP STORAGE (GiB)	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT (IOPS/MBPS)	MAX UNCACHED DISK THROUGHPUT (IOPS/MBPS)	MAX NICs/EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_L 4s	4	32	678	16	20000/200	5000/125	2/4000

SIZE	VCPU	MEMORY (GiB)	TEMP STORAGE (GiB)	MAX DATA DISKS	MAX TEMP STORAGE THROUGHPUT (IOPS/MBPS)	MAX UNCACHED DISK THROUGHPUT (IOPS/MBPS)	MAX NICs/EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_L 8s	8	64	1388	32	40000/400	10000/250	4/8000
Standard_L 16s	16	128	2807	64	80000/800	20000/500	8/16000
Standard_L 32s <sup>1</sup>	32	256	5630	64	160000/1600	40000/1000	8/20000

The maximum disk throughput possible with Ls-series VMs may be limited by the number, size, and striping of any attached disks. For details, see [Design for high performance](#).

<sup>1</sup> Instance is isolated to hardware dedicated to a single customer.

## GS-series

Newer size recommendation: [Easv4-series](#), [Esv4-series](#), [Edsv4-series](#) and M-series

ACU: 180 - 240 <sup>1</sup>

Premium Storage: Supported

Premium Storage caching: Supported

Ephemeral OS Disks: Supported

SIZE	VCPU	MEMORY: GiB	TEMP STORAGE (SSD) GiB	MAX DATA DISKS	MAX CACHED AND TEMP STORAGE THROUGHPUT: IOPS / MBPS (CACHE SIZE IN GiB)	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (Mbps)
Standard_G S1	2	28	56	8	10000/100 (264)	5000/125	2/2000
Standard_G S2	4	56	112	16	20000/200 (528)	10000/250	2/4000
Standard_G S3	8	112	224	32	40000/400 (1056)	20000/500	4/8000
Standard_G S4 <sup>3</sup>	16	224	448	64	80000/800 (2112)	40000/1000	8/16000
Standard_G S5 <sup>2, 3</sup>	32	448	896	64	160000/1600 (4224)	80000/2000	8/20000

<sup>1</sup> The maximum disk throughput (IOPS or MBps) possible with a GS series VM may be limited by the number, size and striping of the attached disk(s). For details, see [Design for high performance](#).

<sup>2</sup> Isolation feature retired on 2/28/2022. For information, see the [retirement announcement](#).

<sup>3</sup> Constrained core sizes available.

## G-series

Newer size recommendation: [Eav4-series](#), [Ev4-series](#) and [Edv4-series](#) and M-series

ACU: 180 - 240

Premium Storage: Not Supported

Premium Storage caching: Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	MAX TEMP STORAGE THROUGHPUT: IOPS/READ MBPS/WRITE MBPS	MAX DATA DISKS/THROUGHPUT: IOPS	MAX NICs/EXPECTED NETWORK BANDWIDTH (MBPS)
Standard_G1	2	28	384	6000/93/46	8/8x500	2/2000
Standard_G2	4	56	768	12000/187/93	16/16x500	2/4000
Standard_G3	8	112	1536	24000/375/187	32/32x500	4/8000
Standard_G4	16	224	3072	48000/750/375	64/64x500	8/16000
Standard_G5 <sup>1</sup>	32	448	6144	96000/1500/750	64/64x500	8/20000

<sup>1</sup> Isolation feature retired on 2/28/2022. For information, see the [retirement announcement](#).

## NV-series

Newer size recommendation: [NVv3-series](#) and [NVv4-series](#)

The NV-series virtual machines are powered by [NVIDIA Tesla M60](#) GPUs and NVIDIA GRID technology for desktop accelerated applications and virtual desktops where customers are able to visualize their data or simulations. Users are able to visualize their graphics intensive workflows on the NV instances to get superior graphics capability and additionally run single precision workloads such as encoding and rendering. NV-series VMs are also powered by Intel Xeon E5-2690 v3 (Haswell) CPUs.

Each GPU in NV instances comes with a GRID license. This license gives you the flexibility to use an NV instance as a virtual workstation for a single user, or 25 concurrent users can connect to the VM for a virtual application scenario.

Premium Storage: Not Supported

Premium Storage caching: Not Supported

Live Migration: Not Supported

Memory Preserving Updates: Not Supported

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICS	VIRTUAL WORKSTATIONS	VIRTUAL APPLICATIONS
Standard_NV6	6	56	340	1	8	24	1	1	25
Standard_NV12	12	112	680	2	16	48	2	2	50
Standard_NV24	24	224	1440	4	32	64	4	4	100

1 GPU = one-half M60 card.

## NC series

Newer size recommendation: [NC T4 v3-series](#)

NC-series VMs are powered by the [NVIDIA Tesla K80](#) card and the Intel Xeon E5-2690 v3 (Haswell) processor. Users can crunch through data faster by leveraging CUDA for energy exploration applications, crash simulations, ray traced rendering, deep learning, and more. The NC24r configuration provides a low latency, high-throughput network interface optimized for tightly coupled parallel computing workloads.

[Premium Storage](#): Not Supported

[Premium Storage caching](#): Not Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX NICS
Standard_NC6	6	56	340	1	12	24	1
Standard_NC12	12	112	680	2	24	48	2
Standard_NC24	24	224	1440	4	48	64	4
Standard_NC24r*	24	224	1440	4	48	64	4

1 GPU = one-half K80 card.

\*RDMA capable

## NCv2 series

Newer size recommendation: [NC T4 v3-series](#) and [NC V100 v3-series](#)

NCv2-series VMs are powered by NVIDIA Tesla P100 GPUs. These GPUs can provide more than 2x the

computational performance of the NC-series. Customers can take advantage of these updated GPUs for traditional HPC workloads such as reservoir modeling, DNA sequencing, protein analysis, Monte Carlo simulations, and others. In addition to the GPUs, the NCv2-series VMs are also powered by Intel Xeon E5-2690 v4 (Broadwell) CPUs.

The NC24rs v2 configuration provides a low latency, high-throughput network interface optimized for tightly coupled parallel computing workloads.

[Premium Storage](#): Supported

[Premium Storage caching](#): Supported

[Live Migration](#): Not Supported

[Memory Preserving Updates](#): Not Supported

[VM Generation Support](#): Generation 1 and 2

[Ephemeral OS Disks](#): Supported

For this VM series, the vCPU (core) quota in your subscription is initially set to 0 in each region. [Request a vCPU quota increase](#) for this series in an [available region](#).

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_NC6s_v2	6	112	736	1	16	12	20000/200	4
Standard_NC12s_v2	12	224	1474	2	32	24	40000/400	8
Standard_NC24s_v2	24	448	2948	4	64	32	80000/800	8
Standard_NC24rs_v2*	24	448	2948	4	64	32	80000/800	8

1 GPU = one P100 card.

\*RDMA capable

## ND series

[Newer size recommendation: NDv2-series and NC V100 v3-series](#)

The ND-series virtual machines are a new addition to the GPU family designed for AI, and Deep Learning workloads. They offer excellent performance for training and inference. ND instances are powered by [NVIDIA Tesla P40](#) GPUs and Intel Xeon E5-2690 v4 (Broadwell) CPUs. These instances provide excellent performance for single-precision floating point operations, for AI workloads utilizing Microsoft Cognitive Toolkit, TensorFlow, Caffe, and other frameworks. The ND-series also offers a much larger GPU memory size (24 GB), enabling to fit much larger neural net models. Like the NC-series, the ND-series offers a configuration with a secondary low-latency, high-throughput network through RDMA, and InfiniBand connectivity so you can run large-scale training jobs spanning many GPUs.

Premium Storage: Supported

Premium Storage caching: Supported

Live Migration: Not Supported

Memory Preserving Updates: Not Supported

VM Generation Support: Generation 1 and 2

Ephemeral OS Disks: Supported

For this VM series, the vCPU (core) quota per region in your subscription is initially set to 0. [Request a vCPU quota increase](#) for this series in an [available region](#).

SIZE	VCPUs	MEMORY: GIB	TEMP STORAGE (SSD) GIB	GPU	GPU MEMORY: GIB	MAX DATA DISKS	MAX UNCACHED DISK THROUGHPUT: IOPS/MBPS	MAX NICs
Standard_ND6s	6	112	736	1	24	12	20000/200	4
Standard_ND12s	12	224	1474	2	48	24	40000/400	8
Standard_ND24s	24	448	2948	4	24	32	80000/800	8
Standard_ND24rs*	24	448	2948	4	96	32	80000/800	8

1 GPU = one P40 card.

\*RDMA capable

## Next steps

Learn more about how [Azure compute units \(ACU\)](#) can help you compare compute performance across Azure SKUs.

# Support for generation 2 VMs on Azure

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Support for generation 2 virtual machines (VMs) is now available on Azure. You can't change a virtual machine's generation after you've created it, so review the considerations on this page before you choose a generation.

Generation 2 VMs support key features that aren't supported in generation 1 VMs. These features include increased memory, Intel Software Guard Extensions (Intel SGX), and virtualized persistent memory (vPMEM). Generation 2 VMs running on-premises, have some features that aren't supported in Azure yet. For more information, see the [Features and capabilities](#) section.

Generation 2 VMs use the new UEFI-based boot architecture rather than the BIOS-based architecture used by generation 1 VMs. Compared to generation 1 VMs, generation 2 VMs might have improved boot and installation times. For an overview of generation 2 VMs and some of the differences between generation 1 and generation 2, see [Should I create a generation 1 or 2 virtual machine in Hyper-V?](#).

## Generation 2 VM sizes

Azure now offers generation 2 support for the following selected VM series:

VM SERIES	GENERATION 1	GENERATION 2
Av2-series	✓	✗
B-series	✓	✓
DCsv2-series	✗	✓
Dv2-series	✓	✗
DSv2-series	✓	✓
Dv3-series	✓	✗
Dsv3-series	✓	✓
Dv4-series	✓	✓
Dsv4-series	✓	✓
Dav4-series	✓	✗
Dasv4-series	✓	✓
Ddv4-series	✓	✓
Ddsv4-series	✓	✓

VM SERIES	GENERATION 1	GENERATION 2
Dasv5-series	✓	✓
Dadsv5-series	✓	✓
DCasv5-series	✗	✓
DCadsv5-series	✗	✓
Dpsv5-series	✗	✓
Dpdsv5-series	✗	✓
Dv5-series	✓	✓
Dsv5-series	✓	✓
Ddv5-series	✓	✓
Ddsv5-series	✓	✓
Ev3-series	✓	✗
Esv3-series	✓	✓
Ev4-series	✓	✗
Esv4-series	✓	✓
Eav4-series	✓	✓
Easv4-series	✓	✓
Edv4-series	✓	✓
Edsv4-series	✓	✓
Easv5-series	✓	✓
Eadsv5-series	✓	✓
ECasv5-series	✗	✓
ECadsv5-series	✗	✓
Epsv5-series	✗	✓
Epdsv5-series	✗	✓
Edv5-series	✓	✓

VM SERIES	GENERATION 1	GENERATION 2
Edsv5-series	✓	✓
Ev5-series	✓	✓
Esv5-series	✓	✓
Fsv2-series	✓	✓
FX-series	✗	✓
GS-series	✗	✓
H-series	✓	✗
HB-series	✓	✓
HBv2-series	✓	✓
HBv3-series	✓	✓
HC-series	✓	✓
Ls-series	✗	✓
Lsv2-series	✓	✓
M-series	✓	✓
Mv2-series <sup>1</sup>	✗	✓
Msv2 and Mdsv2 Medium Memory Series <sup>1</sup>	✗	✓
NC-series	✓	✗
NCv2-series	✓	✓
NCv3-series	✓	✓
NCasT4_v3-series	✓	✓
NC A100 v4-series	✗	✓
ND-series	✓	✓
ND A100 v4-series	✗	✓
NDv2-series	✗	✓
NV-series	✓	✗

VM SERIES	GENERATION 1	GENERATION 2
NVv3-series	✓	✓
NVv4-series	✓	✓
NVadsA10 v5-series	✓	✓
NDm A100 v4-series	✗	✓
NP-series	✓	✗

<sup>1</sup> Mv2-series, DC-series, NDv2-series, Msv2 and Mds2-series Medium Memory do not support Generation 1 VM images and only support a subset of Generation 2 images. Please see [Mv2-series documentation](#), [DSv2-series](#), [ND A100 v4-series](#), [NDv2-series](#), and [Msv2 and Mds2 Medium Memory Series](#) for details.

## Generation 2 VM images in Azure Marketplace

Generation 2 VMs support the following Marketplace images:

- Windows Server 2022, 2019, 2016, 2012 R2, 2012
- Windows 11 Pro, Windows 11 Enterprise
- Windows 10 Pro, Windows 10 Enterprise
- SUSE Linux Enterprise Server 15 SP3, SP2
- SUSE Linux Enterprise Server 12 SP4
- Ubuntu Server 21.04 LTS, 20.04 LTS, 18.04 LTS, 16.04 LTS
- RHEL 8.5, 8.4, 8.3, 8.2, 8.1, 8.0, 7.9, 7.8, 7.7, 7.6, 7.5, 7.4, 7.0
- Cent OS 8.4, 8.3, 8.2, 8.1, 8.0, 7.7, 7.6, 7.5, 7.4
- Oracle Linux 8.4 LVM, 8.3 LVM, 8.2 LVM, 8.1, 7.9 LVM, 7.9, 7.8, 7.7

### NOTE

Specific Virtual machine sizes like Mv2-Series, DC-series, ND A100 v4-series, NDv2-series, Msv2 and Mds2-series may only support a subset of these images - please look at the relevant virtual machine size documentation for complete details.

## On-premises vs. Azure generation 2 VMs

Azure doesn't currently support some of the features that on-premises Hyper-V supports for generation 2 VMs.

GENERATION 2 FEATURE	ON-PREMISES HYPER-V	AZURE
Secure boot	✓	With <a href="#">trusted launch</a>
Shielded VM	✓	✗
vTPM	✓	With <a href="#">trusted launch</a>
Virtualization-based security (VBS)	✓	✓
VHDX format	✓	✗

For more information, see [Trusted launch](#).

## Features and capabilities

### Generation 1 vs. generation 2 features

FEATURE	GENERATION 1	GENERATION 2
Boot	PCAT	UEFI
Disk controllers	IDE	SCSI
VM sizes	All VM sizes	<a href="#">See available sizes</a>

### Generation 1 vs. generation 2 capabilities

CAPABILITY	GENERATION 1	GENERATION 2
OS disk > 2 TB	✗	✓
Custom disk/image/swap OS	✓	✓
Virtual machine scale set support	✓	✓
Azure Site Recovery	✓	✓
Backup/restore	✓	✓
Azure Compute Gallery	✓	✓
Azure disk encryption	✓	✓
Server-side encryption	✓	✓

## Creating a generation 2 VM

### Azure Resource Manager Template

To create a simple Windows Generation 2 VM, see [Create a Windows virtual machine from a Resource Manager template](#) To create a simple Linux Generation 2 VM, see [How to create a Linux virtual machine with Azure Resource Manager templates](#)

### Marketplace image

In the Azure portal or Azure CLI, you can create generation 2 VMs from a Marketplace image that supports UEFI boot.

#### Azure portal

Below are the steps to create a generation 2 (Gen2) VM in Azure portal.

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Search for **Virtual Machines**
3. Under **Services**, select **Virtual machines**.
4. In the **Virtual machines** page, select **Add**, and then select **Virtual machine**.
5. Under **Project details**, make sure the correct subscription is selected.

6. Under **Resource group**, select **Create new** and type a name for your resource group or select an existing resource group from the dropdown.
7. Under **Instance details**, type a name for the virtual machine name and choose a region
8. Under **Image**, select a Gen2 image from the **Marketplace images to get started**

**TIP**

If you don't see the Gen 2 version of the image you want in the drop-down, select **See all images** and then change the **Image Type** filter to **Gen 2**.

9. Select a VM size that supports Gen2. See a list of [supported sizes](#).
10. Fill in the **Administrator account** information and then **Inbound port rules**
11. At the bottom of the page, select **Review + Create**
12. On the **Create a virtual machine** page, you can see the details about the VM you are about to deploy.  
Once validation shows as passed, select **Create**.

**PowerShell**

You can also use PowerShell to create a VM by directly referencing the generation 1 or generation 2 SKU.

For example, use the following PowerShell cmdlet to get a list of the SKUs in the **WindowsServer** offer.

```
Get-AzVMImageSku -Location westus2 -PublisherName MicrosoftWindowsServer -Offer WindowsServer
```

If you're creating a VM with Windows Server 2019 as the OS, then you can select a generation 2 (UEFI) image which looks like this:

```
2019-datacenter-gensecond
```

If you're creating a VM with Windows 10 as the OS, then you can select a generation 2 (UEFI) image which looks like this:

```
20H2-PRO-G2
```

See the [Features and capabilities](#) section for a current list of supported Marketplace images.

**Azure CLI**

Alternatively, you can use the Azure CLI to see any available generation 2 images, listed by **Publisher**.

```
az vm image list --publisher Canonical --sku gen2 --output table --all
```

**Managed image or managed disk**

You can create a generation 2 VM from a managed image or managed disk in the same way you would create a generation 1 VM.

**Virtual machine scale sets**

You can also create generation 2 VMs by using virtual machine scale sets. In the Azure CLI, use Azure scale sets to create generation 2 VMs.

## Frequently asked questions

- Are generation 2 VMs available in all Azure regions?

Yes. But not all [generation 2 VM sizes](#) are available in every region. The availability of the generation 2 VM depends on the availability of the VM size.

- **Is there a price difference between generation 1 and generation 2 VMs?**  
No.
- **I have a .vhf file from my on-premises generation 2 VM. Can I use that .vhf file to create a generation 2 VM in Azure?** Yes, you can bring your generation 2 .vhf file to Azure and use that to create a generation 2 VM. Use the following steps to do so:
  1. Upload the .vhf to a storage account in the same region where you'd like to create your VM.
  2. Create a managed disk from the .vhf file. Set the Hyper-V Generation property to V2. The following PowerShell commands set Hyper-V Generation property when creating managed disk.

```
$sourceUri = 'https://xyzstorage.blob.core.windows.net/vhd/abcd.vhd'. #<Provide location to  
your uploaded .vhf file>  
$osDiskName = 'gen2Diskfrmenvhd' #<Provide a name for your disk>  
$diskconfig = New-AzDiskConfig -Location '<location>' -DiskSizeGB 127 -AccountType  
Standard_LRS -OsType Windows -HyperVGeneration "V2" -SourceUri $sourceUri -CreateOption  
'Import'  
New-AzDisk -DiskName $osDiskName -ResourceGroupName '<Your Resource Group>' -Disk $diskconfig
```

- 3. Once the disk is available, create a VM by attaching this disk. The VM created will be a generation 2 VM. When the generation 2 VM is created, you can optionally generalize the image of this VM. By generalizing the image, you can use it to create multiple VMs.
- **How do I increase the OS disk size?**

OS disks larger than 2 TiB are new to generation 2 VMs. By default, OS disks are smaller than 2 TiB for generation 2 VMs. You can increase the disk size up to a recommended maximum of 4 TiB. Use the Azure CLI or the Azure portal to increase the OS disk size. For information about how to expand disks programmatically, see [Resize a disk](#) for [Windows](#) or [Linux](#).

To increase the OS disk size from the Azure portal:

1. In the Azure portal, go to the VM properties page.
2. To shut down and deallocate the VM, select the **Stop** button.
3. In the **Disk** section, select the OS disk you want to increase.
4. In the **Disk** section, select **Configuration**, and update the **Size** to the value you want.
5. Go back to the VM properties page and **Start** the VM.

You might see a warning for OS disks larger than 2 TiB. The warning doesn't apply to generation 2 VMs. However, OS disk sizes larger than 4 TiB are not supported.

- **Do generation 2 VMs support accelerated networking?**  
Yes. For more information, see [Create a VM with accelerated networking](#).
- **Do generation 2 VMs support Secure Boot or vTPM in Azure?** Both vTPM and Secure Boot are features of trusted launch for generation 2 VMs. For more information, see [Trusted launch](#).
- **Is VHDX supported on generation 2?**  
No, generation 2 VMs support only VHD.
- **Do generation 2 VMs support Azure Ultra Disk Storage?**  
Yes.
- **Can I migrate a VM from generation 1 to generation 2?**  
No, you can't change the generation of a VM after you create it. If you need to switch between VM

generations, create a new VM of a different generation.

- **Why is my VM size not enabled in the size selector when I try to create a Gen2 VM?**

This may be solved by doing the following:

1. Verify that the **VM generation** property is set to **Gen 2**.
2. Verify you are searching for a **VM size which supports Gen2 VMs**.

## Next steps

Learn more about the [trusted launch](#) with gen 2 VMs.

Learn about [generation 2 virtual machines in Hyper-V](#).

# Virtual machine isolation in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure Compute offers virtual machine sizes that are Isolated to a specific hardware type and dedicated to a single customer. The Isolated sizes live and operate on specific hardware generation and will be deprecated when the hardware generation is retired.

Isolated virtual machine sizes are best suited for workloads that require a high degree of isolation from other customers' workloads for reasons that include meeting compliance and regulatory requirements. Utilizing an isolated size guarantees that your virtual machine will be the only one running on that specific server instance.

Additionally, as the Isolated size VMs are large, customers may choose to subdivide the resources of these VMs by using [Azure support for nested virtual machines](#).

The current Isolated virtual machine offerings include:

- Standard\_E80ids\_v4
- Standard\_E80is\_v4
- Standard\_E104i\_v5
- Standard\_E104is\_v5
- Standard\_E104id\_v5
- Standard\_E104ids\_v5
- Standard\_M192is\_v2
- Standard\_M192ims\_v2
- Standard\_M192ids\_v2
- Standard\_M192idms\_v2
- Standard\_F72s\_v2
- Standard\_M128ms

## NOTE

Isolated VM Sizes have a hardware limited lifespan. Please see below for details

## Deprecation of Isolated VM Sizes

Isolated VM sizes have a hardware limited lifespan. Azure will issue reminders 12 months in advance of the official deprecation date of the sizes and will provide an updated isolated offering for your consideration.

SIZE	ISOLATION RETIREMENT DATE
Standard_DS15_v2	May 15, 2021
Standard_D15_v2	May 15, 2021
Standard_G5	February 15, 2022
Standard_GS5	February 15, 2022

SIZE	ISOLATION RETIREMENT DATE
Standard_E64i_v3	February 15, 2022
Standard_E64is_v3	February 15, 2022

## FAQ

**Q: Is the size going to get retired or only its "isolation" feature?**

A: Currently, only the isolation feature of the VM sizes is being retired. The deprecated isolated sizes will continue to exist in non-isolated state. If isolation is not needed, there is no action to be taken and the VM will continue to work as expected.

**Q: Is there a downtime when my vm lands on a non-isolated hardware?**

A: If there is no need of isolation, no action is needed and there will be no downtime. On contrary if isolation is required, our announcement will include the recommended replacement size. Selecting the replacement size will require our customers to resize their VMs.

**Q: Is there any cost delta for moving to a non-isolated virtual machine?**

A: No

**Q: When are the other isolated sizes going to retire?**

A: We will provide reminders 12 months in advance of the official deprecation of the isolated size. Our latest announcement includes isolation feature retirement of Standard\_G5, Standard\_GS5, Standard\_E64i\_v3 and Standard\_E64i\_v3.

**Q: I'm an Azure Service Fabric Customer relying on the Silver or Gold Durability Tiers. Does this change impact me?**

A: No. The guarantees provided by Service Fabric's [Durability Tiers](#) will continue to function even after this change. If you require physical hardware isolation for other reasons, you may still need to take one of the actions described above.

**Q: What are the milestones for D15\_v2 or DS15\_v2 isolation retirement?**

A:

DATE	ACTION
May 15, 2020 <sup>1</sup>	D/DS15_v2 isolation retirement announcement
May 15, 2021	D/DS15_v2 isolation guarantee removed

<sup>1</sup> Existing customer using these sizes will receive an announcement email with detailed instructions on the next steps.

**Q: What are the milestones for G5, Gs5, E64i\_v3 and E64is\_v3 isolation retirement?**

A:

DATE	ACTION
Feb 15, 2021 <sup>1</sup>	G5/GS5/E64i_v3/E64is_v3 isolation retirement announcement
Feb 28, 2022	G5/GS5/E64i_v3/E64is_v3 isolation guarantee removed

<sup>1</sup> Existing customer using these sizes will receive an announcement email with detailed instructions on the next steps.

## Next steps

Customers can also choose to further subdivide the resources of these Isolated virtual machines by using [Azure support for nested virtual machines](#).

# Azure compute unit (ACU)

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The concept of the Azure Compute Unit (ACU) provides a way of comparing compute (CPU) performance across Azure SKUs. This will help you easily identify which SKU is most likely to satisfy your performance needs. ACU is currently standardized on a Small (Standard\_A1) VM being 100 and all other SKUs then represent approximately how much faster that SKU can run a standard benchmark.

\*ACUs use Intel® Turbo technology to increase CPU frequency and provide a performance increase. The amount of the performance increase can vary based on the VM size, workload, and other workloads running on the same host.

\*\*ACUs use AMD® Boost technology to increase CPU frequency and provide a performance increase. The amount of the performance increase can vary based on the VM size, workload, and other workloads running on the same host.

\*\*\*Hyper-threaded and capable of running nested virtualization

\*\*\*\*AMD Simultaneous multithreading technology

## IMPORTANT

The ACU is only a guideline. The results for your workload may vary.

SKU FAMILY	ACU \ VCPU	VCPU: CORE
A1_v2 - A8_v2	100	1:1
A2m_v2 - A8m_v2	100	1:1
B	Varies	1:1
D1 - D14	160 - 250	1:1
D1_v2 - D15_v2	210 - 250*	1:1
DS1 - DS14	160 - 250	1:1
DS1_v2 - DS15_v2	210 - 250*	1:1
D_v3	160 - 190*	2:1***
Ds_v3	160 - 190*	2:1***
Dav4	230 - 260**	2:1****
Dasv4	230 - 260**	2:1****

SKU FAMILY	ACU \ VCPU	VCPU: CORE
Dv4	195 - 210	2:1***
Dsv4	195 - 210	2:1***
Ddv4	195 -210*	2:1***
Ddsv4	195 - 210*	2:1***
E_v3	160 - 190*	2:1***
Es_v3	160 - 190*	2:1***
Eav4	230 - 260**	2:1****
Easv4	230 - 260**	2:1****
Ev4	195 - 210	2:1***
Esv4	195 - 210	2:1***
Edv4	195 - 210*	2:1***
Edsv4	195 - 210*	2:1***
F2s_v2 - F72s_v2	195 - 210*	2:1***
F1 - F16	210 - 250*	1:1
F1s - F16s	210 - 250*	1:1
FX4 - FX48	310 - 340*	2:1***
G1 - G5	180 - 240*	1:1
GS1 - GS5	180 - 240*	1:1
H	290 - 300*	1:1
HB	199 - 216**	1:1
HC	297 - 315*	1:1
L4s - L32s	180 - 240*	1:1
L8s_v2 - L80s_v2	150 - 175**	2:1****
M	160 - 180	2:1***
Mv2	240 - 280	2:1***

SKU FAMILY	ACU \ VCPU	VCPU: CORE
NVv4	230 - 260**	2:1****

Processor model information for each SKU is available in the SKU documentation (see links above). Optimal performance may require the latest VM images (OS and [VM generation](#)) to ensure the latest updates and fastest drivers.

## VM Series Retiring

The following VM series are retiring on or before August 31, 2024:

SKU FAMILY	ACU \ VCPU	VCPU: CORE	RETIREMENT DATE
H	290 - 300*	1:1	<a href="#">August 31, 2022</a>
HB	199 - 216**	1:1	<a href="#">August 31, 2024</a>
A0	50	1:1	<a href="#">August 31, 2024</a>
A1 - A4	100	1:1	<a href="#">August 31, 2024</a>
A5 - A7	100	1:1	<a href="#">August 31, 2024</a>
A8 - A11	225*	1:1	<a href="#">August 31, 2024</a>

The following GPU series are also retiring:

SKU FAMILY	RETIREMENT DATE
NC	<a href="#">August 31, 2023</a>
NCv2	<a href="#">August 31, 2023</a>
ND	<a href="#">August 31, 2023</a>
NV	<a href="#">August 31, 2023</a>

## Performance Consistency

We understand that Azure customers want the best possible consistent performance, they want to be able to count on getting the same performance from the same type of VM every time.

Azure VM sizes typically run with maximum performance on the hardware platform they are first released on. Azure may place controls on older Azure VMs when run on newer hardware to help maintain consistent performance for our customers even when the VMs run on different hardware. For example:

1. D, E, and F series VMs may have the processor frequency set to a lower level when running on newer hardware to help achieve better performance consistency across hardware updates. (The specific frequency setting varies based on the processor the VM series was first released on and the comparable performance of the current hardware.)
2. A series VMs use an older model based on time slicing newer hardware to deliver performance consistency across hardware versions.
3. B series VMs are burstable and use a credit system (described in their [documentation](#)) to achieve expected

performance.

These different processor settings for VMs are a key part of Azure's effort to provide consistent performance and minimize the impact of changes in underlying hardware platform outside of our customer's control.

## More Info

Here are links to more information about the different sizes:

- [General-purpose](#)
- [Memory optimized](#)
- [Compute optimized](#)
- [GPU optimized](#)
- [High performance compute](#)
- [Storage optimized](#)

# Compute benchmark scores for Linux VMs

9/21/2022 • 55 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The following CoreMark benchmark scores show compute performance for Azure's high-performance VM lineup running Ubuntu 18.04. Compute benchmark scores are also available for [Windows VMs](#).

## Azure (Coremark) TOC

TYPE	FAMILIES
Compute optimized	Fsv2, FXMDVS
General purpose	B, DADSv5, DASv5, DDSv5, DDv5, DSv5, Dv5, Dasv4, Dav4, Ddsv4, Ddv4, Dsv4, Dv4, Dsv3, Dv3, DSv2, Dv2
High performance compute	HBv3, HBv2, HB, HC
Memory optimized	EADSv5, EASv5, EDSv5, EDv5, ESv5, Ev5, Easv4, Eav4, Edsv4, Edv4, Esv4, Ev4, DSv2, Esv3, Ev3, Dv2, Msv2, Ms
Storage optimized	Lsv2
Confidential Compute	DCS, DCSv3, DCDSv3, DCv2

## Compute optimized

### Fsv2 - Compute + Premium Storage

(10/10/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F2s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	2	1	4.0	35,925	603	1.68%	308
Standard_F2s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	4.0	35,659	344	0.96%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F4s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	4	1	8.0	65,819	851	1.29%	245
Standard_F4s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	8.0	65,683	459	0.70%	98
Standard_F8s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	8	1	16.0	136,027	1,272	0.94%	238
Standard_F8s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	16.0	135,829	912	0.67%	91
Standard_F16s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	16	1	32.0	271,369	1,955	0.72%	280
Standard_F16s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	32.0	271,236	1,325	0.49%	105
Standard_F32s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	32	1	64.0	538,734	3,795	0.70%	245
Standard_F32s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	64.0	537,915	7,646	1.42%	91

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F48s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	48	2	96.0	780,596	8,712	1.12%	182
Standard_F48s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	96.0	749,662	21,751	2.90%	56
Standard_F64s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	64	2	128.0	1,035,424	11,557	1.12%	175
Standard_F64s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	128.0	1,023,867	17,292	1.69%	63
Standard_F72s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	72	2	144.0	1,126,078	12,094	1.07%	182
Standard_F72s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	72	2	144.0	1,120,116	15,662	1.40%	42

## FXMDVS

(04/12/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_FX4mds	Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz	4	1	82.5	77,794	711	0.91%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_FX12mds	Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz	12	1	247.8	228,808	827	0.36%	42
Standard_FX24mds	Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz	24	1	495.8	475,203	2,142	0.45%	42
Standard_FX36mds	Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz	36	2	744.1	678,947	10,647	1.57%	42
Standard_FX48mds	Intel(R) Xeon(R) Gold 6246R CPU @ 3.40GHz	48	2	992.1	899,165	12,000	1.33%	42

## General purpose

### B - Burstable

(09/24/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B1s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	1.0	18,768	429	2.29%	35
Standard_B1s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	1.0	19,725	1,258	6.38%	112
Standard_B1s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	1.0	18,287	2,316	12.66%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B1s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	1.0	21,992	371	1.69%	21
Standard_B1ls	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	0.5	19,162	605	3.16%	28
Standard_B1ls	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	0.5	19,365	810	4.18%	126
Standard_B1ls	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	0.5	18,780	2,840	15.12%	63
Standard_B1ls	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	0.5	21,954	461	2.10%	21
Standard_B1ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	2.0	18,167	82	0.45%	14
Standard_B1ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	2.0	20,113	1,188	5.90%	126
Standard_B1ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	2.0	18,882	2,434	12.89%	77

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B1ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	2.0	22,182	57	0.26%	21
Standard_B2s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	4.0	36,074	431	1.20%	21
Standard_B2s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	4.0	37,634	1,445	3.84%	119
Standard_B2s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	4.0	37,824	5,913	15.63%	70
Standard_B2s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	4.0	44,147	397	0.90%	28
Standard_B2ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	8.0	36,338	333	0.92%	28
Standard_B2ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	8.0	38,026	1,749	4.60%	119
Standard_B2ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	8.0	38,060	5,978	15.71%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B2ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	44,227	349	0.79%	21
Standard_B4ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	16.0	70,362	3,310	4.70%	28
Standard_B4ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	16.0	74,405	2,647	3.56%	112
Standard_B4ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	16.0	71,174	8,008	11.25%	77
Standard_B4ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	85,894	1,445	1.68%	21
Standard_B8ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	32.0	144,753	3,091	2.14%	35
Standard_B8ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	32.0	145,027	1,460	1.01%	98
Standard_B8ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	32.0	138,848	11,991	8.64%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B8ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	170,427	2,402	1.41%	14
Standard_B12ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	12	1	48.0	214,731	4,672	2.18%	28
Standard_B12ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	12	1	48.0	219,383	2,213	1.01%	112
Standard_B12ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	12	1	48.0	202,912	17,172	8.46%	77
Standard_B12ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	12	1	48.0	256,373	1,768	0.69%	21
Standard_B16ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	64.0	276,669	8,921	3.22%	35
Standard_B16ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	64.0	294,079	3,437	1.17%	112
Standard_B16ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	64.0	270,563	27,650	10.22%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B16ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	340,582	6,378	1.87%	21
Standard_B20ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	80.0	347,156	5,967	1.72%	14
Standard_B20ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	80.0	363,209	3,837	1.06%	112
Standard_B20ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	80.0	342,212	21,775	6.36%	91
Standard_B20ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	80.0	416,103	3,337	0.80%	21

## DADSv5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2ads_v5	AMD EPYC 7763 64-Core Processor	2	1	7.8	38,919	41	0.10%	35
Standard_D4ads_v5	AMD EPYC 7763 64-Core Processor	4	1	15.6	72,644	172	0.24%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D8ads_v5	AMD EPYC 7763 64-Core Processor	8	1	31.4	153,951	552	0.36%	35
Standard_D16ads_v5	AMD EPYC 7763 64-Core Processor	16	1	62.8	306,800	941	0.31%	35
Standard_D32ads_v5	AMD EPYC 7763 64-Core Processor	32	1	125.8	600,925	7,721	1.28%	35
Standard_D48ads_v5	AMD EPYC 7763 64-Core Processor	48	1	188.7	893,740	12,161	1.36%	35
Standard_D64ads_v5	AMD EPYC 7763 64-Core Processor	64	1	251.7	1,195,169	16,506	1.38%	35
Standard_D96ads_v5	AMD EPYC 7763 64-Core Processor	96	2	377.9	1,831,129	20,839	1.14%	35

## DASv5

(03/14/2022 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2as_v5	AMD EPYC 7763 64-Core Processor	2	1	7.8	38,869	118	0.30%	35
Standard_D4as_v5	AMD EPYC 7763 64-Core Processor	4	1	15.6	72,928	460	0.63%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D8as_v5	AMD EPYC 7763 64-Core Processor	8	1	31.4	153,842	462	0.30%	35
Standard_D16as_v5	AMD EPYC 7763 64-Core Processor	16	1	62.8	304,560	3,347	1.10%	35
Standard_D32as_v5	AMD EPYC 7763 64-Core Processor	32	1	125.8	599,269	8,844	1.48%	35
Standard_D48as_v5	AMD EPYC 7763 64-Core Processor	48	1	188.7	896,034	12,918	1.44%	35
Standard_D64as_v5	AMD EPYC 7763 64-Core Processor	64	1	251.7	1,195,829	16,444	1.38%	35
Standard_D96as_v5	AMD EPYC 7763 64-Core Processor	96	2	377.9	1,833,797	20,117	1.10%	35

## DDSv5

(03/14/2022 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	7.8	34,926	11	0.03%	56
Standard_D4ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	15.6	68,673	207	0.30%	56

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D8ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	31.4	136,764	489	0.36%	49
Standard_D16ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	62.8	273,303	1,122	0.41%	56
Standard_D32ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	125.8	545,658	2,409	0.44%	49
Standard_D48ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	1	188.7	813,359	5,923	0.73%	49
Standard_D64ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	251.9	1,061,667	10,151	0.96%	35
Standard_D96ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	377.9	1,577,187	17,287	1.10%	56

## DDv5

(03/11/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	7.8	34,923	16	0.05%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	15.6	68,696	234	0.34%	56
Standard_D8d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	31.4	136,791	496	0.36%	42
Standard_D16d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	62.8	273,463	1,085	0.40%	49
Standard_D32d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	125.8	544,718	2,672	0.49%	49
Standard_D48d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	1	188.7	812,195	7,150	0.88%	56
Standard_D64d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	251.9	1,061,317	11,637	1.10%	42
Standard_D96d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	377.9	1,579,691	19,962	1.26%	49

**DSv5**

(03/14/2022 PBID:7668456 )

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	7.8	32,093	2,789	8.69%	63
Standard_D4s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	15.6	67,114	3,816	5.69%	42
Standard_D8s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	31.4	129,302	9,365	7.24%	49
Standard_D16s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	62.8	250,482	19,982	7.98%	49
Standard_D32s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	125.8	500,612	33,756	6.74%	42
Standard_D48s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	1	188.7	725,001	39,805	5.49%	35
Standard_D64s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	251.9	965,147	62,581	6.48%	42
Standard_D96s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	377.9	1,422,950	79,151	5.56%	35

Dv5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	7.8	29,597	565	1.91%	49
Standard_D4_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	15.6	66,338	3,932	5.93%	56
Standard_D8_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	31.4	121,070	8,218	6.79%	245
Standard_D16_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	62.8	247,579	18,327	7.40%	42
Standard_D32_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	125.8	475,867	11,659	2.45%	35
Standard_D48_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	1	188.7	733,540	43,528	5.93%	35
Standard_D64_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	251.9	924,146	9,983	1.08%	35
Standard_D96_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	377.9	1,378,842	14,644	1.06%	35

**Dasv4**

(09/23/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2as_v4	AMD EPYC 7452 32-Core Processor	2	1	8.0	38,230	313	0.82%	77
Standard_D4as_v4	AMD EPYC 7452 32-Core Processor	4	1	16.0	71,707	790	1.10%	98
Standard_D8as_v4	AMD EPYC 7452 32-Core Processor	8	1	32.0	151,474	1,599	1.06%	56
Standard_D16as_v4	AMD EPYC 7452 32-Core Processor	16	2	64.0	292,074	9,782	3.35%	91
Standard_D32as_v4	AMD EPYC 7452 32-Core Processor	32	4	128.0	575,465	14,574	2.53%	98
Standard_D48as_v4	AMD EPYC 7452 32-Core Processor	48	6	192.0	843,316	21,739	2.58%	105
Standard_D64as_v4	AMD EPYC 7452 32-Core Processor	64	8	256.0	1,120,467	30,714	2.74%	7

**Dav4**

(09/23/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2a_v4	AMD EPYC 7452 32-Core Processor	2	1	8.0	38,061	1,219	3.20%	119

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4a_v4	AMD EPYC 7452 32-Core Processor	4	1	16.0	70,996	2,280	3.21%	140
Standard_D8a_v4	AMD EPYC 7452 32-Core Processor	8	1	32.0	151,379	1,946	1.29%	147
Standard_D16a_v4	AMD EPYC 7452 32-Core Processor	16	2	64.0	293,654	6,263	2.13%	161
Standard_D32a_v4	AMD EPYC 7452 32-Core Processor	32	4	128.0	571,897	16,009	2.80%	147
Standard_D48a_v4	AMD EPYC 7452 32-Core Processor	48	6	192.0	837,332	22,923	2.74%	161
Standard_D64a_v4	AMD EPYC 7452 32-Core Processor	64	8	256.0	1,126,793	29,148	2.59%	21
Standard_D96a_v4	AMD EPYC 7452 32-Core Processor	96	12	384.0	1,590,434	31,887	2.00%	14

## DDSV4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	35,557	740	2.08%	189

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	65,958	788	1.19%	189
Standard_D8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	135,907	1,108	0.81%	189
Standard_D16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	271,137	1,374	0.51%	189
Standard_D32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	540,212	4,954	0.92%	189
Standard_D48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	756,538	15,048	1.99%	154
Standard_D48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	775,291	11,776	1.52%	35
Standard_D64ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,025,017	14,482	1.41%	182

#### DDv4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	35,640	371	1.04%	189
Standard_D4d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	65,767	590	0.90%	189
Standard_D8d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	135,756	901	0.66%	189
Standard_D16d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	271,216	1,630	0.60%	189
Standard_D32d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	540,052	4,426	0.82%	189
Standard_D48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	744,246	20,864	2.80%	147
Standard_D48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	769,702	10,800	1.40%	42
Standard_D64d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,021,118	15,896	1.56%	189

Dsv4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	35,794	578	1.61%	175
Standard_D4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	65,910	668	1.01%	182
Standard_D8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	135,542	835	0.62%	182
Standard_D16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	271,053	1,429	0.53%	182
Standard_D32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	539,271	8,236	1.53%	182
Standard_D48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	743,539	18,204	2.45%	140
Standard_D48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	772,877	9,466	1.22%	35
Standard_D64s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,024,799	15,499	1.51%	182

**Dv4**

(09/21/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	35,651	228	0.64%	182
Standard_D4_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	65,896	647	0.98%	175
Standard_D8_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	135,803	963	0.71%	182
Standard_D16_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	271,203	1,345	0.50%	182
Standard_D32_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	539,406	5,745	1.07%	182
Standard_D48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	745,137	17,607	2.36%	154
Standard_D48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	771,269	7,936	1.03%	28

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D64_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,020,266	19,814	1.94%	182

### DSv3 - General Compute + Premium Storage

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	8.0	26,323	1,194	4.54%	35
Standard_D2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	8.0	27,724	2,091	7.54%	84
Standard_D2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	8.0	26,916	1,424	5.29%	91
Standard_D2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	31,342	18	0.06%	21
Standard_D4s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	16.0	50,716	1,058	2.09%	35
Standard_D4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	16.0	52,445	3,515	6.70%	77

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	16.0	50,029	3,332	6.66%	91
Standard_D4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	57,991	277	0.48%	35
Standard_D8s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	32.0	99,350	179	0.18%	7
Standard_D8s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	32.0	101,053	7,572	7.49%	126
Standard_D8s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	32.0	102,485	3,143	3.07%	56
Standard_D8s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	120,003	546	0.46%	35
Standard_D16s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	1	64.0	199,434	1,886	0.95%	35
Standard_D16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	64.0	196,242	5,336	2.72%	77

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	64.0	200,086	3,164	1.58%	98
Standard_D16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	239,837	1,457	0.61%	28
Standard_D32s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	32	2	128.0	389,200	3,438	0.88%	35
Standard_D32s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	1	128.0	390,983	1,845	0.47%	91
Standard_D32s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	128.0	397,239	2,996	0.75%	77
Standard_D32s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	478,885	1,968	0.41%	35
Standard_D48s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	48	2	192.0	571,460	4,725	0.83%	112
Standard_D48s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	1	192.0	581,930	3,357	0.58%	63

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D48s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	2	192.0	578,181	5,736	0.99%	35
Standard_D48s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	698,208	8,889	1.27%	28
Standard_D64s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	64	2	256.0	761,965	4,971	0.65%	49
Standard_D64s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	64	2	256.0	767,645	10,497	1.37%	154
Standard_D64s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	909,271	7,791	0.86%	28

### Dv3 - General Compute

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	8.0	25,478	977	3.84%	91
Standard_D2_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	8.0	26,601	1,381	5.19%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	8.0	27,116	1,688	6.23%	70
Standard_D2_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	31,352	22	0.07%	7
Standard_D4_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	16.0	49,708	2,668	5.37%	77
Standard_D4_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	16.0	51,650	3,799	7.35%	91
Standard_D4_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	16.0	50,025	2,210	4.42%	70
Standard_D8_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	32.0	100,422	1,323	1.32%	98
Standard_D8_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	32.0	97,861	5,903	6.03%	91
Standard_D8_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	32.0	100,826	1,597	1.58%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D8_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	120,063	559	0.47%	7
Standard_D16_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	1	64.0	198,362	2,228	1.12%	84
Standard_D16_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	64.0	193,498	2,080	1.07%	77
Standard_D16_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	64.0	199,683	1,862	0.93%	63
Standard_D16_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	239,145	1,767	0.74%	7
Standard_D32_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	32	2	128.0	387,724	3,770	0.97%	63
Standard_D32_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	1	128.0	388,466	9,936	2.56%	112
Standard_D32_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	128.0	397,605	2,423	0.61%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D32_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	479,158	1,356	0.28%	14
Standard_D48_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	48	2	192.0	569,331	6,445	1.13%	140
Standard_D48_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	1	192.0	578,441	6,615	1.14%	70
Standard_D48_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	2	192.0	576,459	7,500	1.30%	21
Standard_D48_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	693,147	8,980	1.30%	7
Standard_D64_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	64	2	256.0	762,667	5,489	0.72%	42
Standard_D64_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	64	2	256.0	766,739	7,516	0.98%	189
Standard_D64_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	916,549	8,944	0.98%	7

#### DSv2 - General Purpose + Premium Storage

(09/30/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	3.5	18,952	876	4.62%	49
Standard_DS1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	3.5	20,066	937	4.67%	70
Standard_DS1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	3.5	19,721	1,686	8.55%	98
Standard_DS1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	3.5	22,210	25	0.11%	14
Standard_DS2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	7.0	37,635	977	2.60%	70
Standard_DS2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	7.0	39,784	2,446	6.15%	84
Standard_DS2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	7.0	37,761	1,322	3.50%	70
Standard_DS2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	7.0	44,728	39	0.09%	7

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS3_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	14.0	73,128	1,720	2.35%	70
Standard_DS3_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	14.0	75,583	4,279	5.66%	70
Standard_DS3_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	14.0	72,459	1,230	1.70%	91
Standard_DS4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	28.0	146,804	1,270	0.86%	49
Standard_DS4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	28.0	147,510	3,214	2.18%	98
Standard_DS4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	28.0	144,002	1,731	1.20%	70
Standard_DS4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	28.0	116,687	59,050	50.61%	14
Standard_DS5_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	56.0	284,713	4,702	1.65%	63

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS5_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	56.0	297,550	1,502	0.50%	70
Standard_DS5_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	56.0	287,238	4,723	1.64%	84
Standard_DS5_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	56.0	345,144	2,027	0.59%	14

## Dv2 - General Compute

(09/30/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	3.5	19,112	851	4.45%	105
Standard_D1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	3.5	21,866	1,802	8.24%	49
Standard_D1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	3.5	19,641	808	4.11%	63
Standard_D2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	7.0	36,684	1,542	4.20%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	7.0	40,452	2,922	7.22%	42
Standard_D2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	7.0	37,109	1,525	4.11%	63
Standard_D3_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	14.0	72,681	1,424	1.96%	84
Standard_D3_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	14.0	76,631	4,948	6.46%	70
Standard_D3_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	14.0	73,768	2,279	3.09%	56
Standard_D4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	28.0	141,204	19,309	13.67%	70
Standard_D4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	28.0	149,027	1,783	1.20%	98
Standard_D4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	28.0	143,576	1,305	0.91%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D5_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	56.0	281,644	9,594	3.41%	105
Standard_D5_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	56.0	296,879	2,069	0.70%	28
Standard_D5_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	56.0	284,996	5,417	1.90%	77

## High performance compute

### HBv3

(04/02/2022 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HB120rs_v3	AMD EPYC 7V13 64-Core Processor	120	4	440.9	2,725,819	62,125	2.28%	7
Standard_HB120rs_v3	AMD EPYC 7V73X 64-Core Processor	120	4	440.9	2,736,592	88,258	3.23%	49
Standard_HB120-16rs_v3	AMD EPYC 7V73X 64-Core Processor	16	4	440.9	441,414	16,866	3.82%	35
Standard_HB120-32rs_v3	AMD EPYC 7V73X 64-Core Processor	32	4	440.9	821,802	31,550	3.84%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HB120-32rs_v3	AMD EPYC 7V13 64-Core Processor	32	4	440.9	812,130	29,738	3.66%	7
Standard_HB120-64rs_v3	AMD EPYC 7V73X 64-Core Processor	64	4	440.9	1,602,181	77,107	4.81%	42
Standard_HB120-96rs_v3	AMD EPYC 7V73X 64-Core Processor	96	4	440.9	2,396,063	79,104	3.30%	35
Standard_HB120-96rs_v3	AMD EPYC 7V13 64-Core Processor	96	4	440.9	2,345,787	74,660	3.18%	7

## HBrsv2

(04/02/2022 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HB120rs_v2	AMD EPYC 7V12 64-Core Processor	120	30	425.1	2,583,980	68,594	2.65%	133

## HBS - memory bandwidth (AMD EPYC)

(10/14/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HB60rs	AMD EPYC 7551 32-Core Processor	60	15	228.0	986,593	28,102	2.85%	21

## HCS - dense computation (Intel Xeon Platinum 8168)

(10/14/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HC44rs	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	44	2	352.0	995,006	25,995	2.61%	21

## Memory optimized

### EADSV5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2ads_v5	AMD EPYC 7763 64-Core Processor	2	1	15.6	38,922	47	0.12%	35
Standard_E4ads_v5	AMD EPYC 7763 64-Core Processor	4	1	31.4	72,638	140	0.19%	35
Standard_E4-2ads_v5	AMD EPYC 7763 64-Core Processor	2	1	31.4	38,924	52	0.13%	35
Standard_E8ads_v5	AMD EPYC 7763 64-Core Processor	8	1	62.8	153,765	572	0.37%	35
Standard_E8-2ads_v5	AMD EPYC 7763 64-Core Processor	2	1	62.8	38,916	48	0.12%	35
Standard_E8-4ads_v5	AMD EPYC 7763 64-Core Processor	4	1	62.8	73,154	920	1.26%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16ads_v5	AMD EPYC 7763 64-Core Processor	16	1	125.8	303,780	4,336	1.43%	35
Standard_E16-4ads_v5	AMD EPYC 7763 64-Core Processor	4	1	125.8	72,871	365	0.50%	35
Standard_E16-8ads_v5	AMD EPYC 7763 64-Core Processor	8	1	125.8	153,767	603	0.39%	35
Standard_E20ads_v5	AMD EPYC 7763 64-Core Processor	20	2	157.4	375,535	4,506	1.20%	35
Standard_E32ads_v5	AMD EPYC 7763 64-Core Processor	32	1	251.7	599,632	7,739	1.29%	35
Standard_E32-8ads_v5	AMD EPYC 7763 64-Core Processor	8	1	251.7	153,859	642	0.42%	35
Standard_E32-16ads_v5	AMD EPYC 7763 64-Core Processor	16	1	251.7	305,170	3,524	1.15%	35
Standard_E48ads_v5	AMD EPYC 7763 64-Core Processor	48	1	377.7	892,509	12,866	1.44%	35
Standard_E64ads_v5	AMD EPYC 7763 64-Core Processor	64	1	503.7	1,195,479	15,280	1.28%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E64-16ads_v5	AMD EPYC 7763 64-Core Processor	16	1	503.7	307,095	958	0.31%	35
Standard_E64-32ads_v5	AMD EPYC 7763 64-Core Processor	32	1	503.7	598,891	7,892	1.32%	35
Standard_E96ads_v5	AMD EPYC 7763 64-Core Processor	96	2	661.4	1,832,942	15,387	0.84%	35
Standard_E96-24ads_v5	AMD EPYC 7763 64-Core Processor	24	1	661.0	450,787	5,791	1.28%	35
Standard_E96-48ads_v5	AMD EPYC 7763 64-Core Processor	48	1	661.0	894,161	10,898	1.22%	35
Standard_E96iads_v5	AMD EPYC 7763 64-Core Processor	96	2	661.4	1,837,643	18,009	0.98%	35
Standard_E112iads_v5	AMD EPYC 7763 64-Core Processor	112	2	661.4	2,125,685	28,945	1.36%	35

## EASv5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2as_v5	AMD EPYC 7763 64-Core Processor	2	1	15.6	38,919	45	0.11%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E4as_v5	AMD EPYC 7763 64-Core Processor	4	1	31.4	72,704	161	0.22%	35
Standard_E4-2as_v5	AMD EPYC 7763 64-Core Processor	2	1	31.4	37,346	3,168	8.48%	35
Standard_E8as_v5	AMD EPYC 7763 64-Core Processor	8	1	62.8	153,881	485	0.31%	35
Standard_E8-2as_v5	AMD EPYC 7763 64-Core Processor	2	1	62.8	38,929	47	0.12%	35
Standard_E8-4as_v5	AMD EPYC 7763 64-Core Processor	4	1	62.8	72,735	153	0.21%	35
Standard_E16as_v5	AMD EPYC 7763 64-Core Processor	16	1	125.8	305,729	2,717	0.89%	35
Standard_E16-4as_v5	AMD EPYC 7763 64-Core Processor	4	1	125.8	72,736	225	0.31%	35
Standard_E16-8as_v5	AMD EPYC 7763 64-Core Processor	8	1	125.8	154,340	805	0.52%	35
Standard_E20as_v5	AMD EPYC 7763 64-Core Processor	20	2	157.4	376,978	4,605	1.22%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32as_v5	AMD EPYC 7763 64-Core Processor	32	1	251.7	595,261	7,046	1.18%	35
Standard_E32-8as_v5	AMD EPYC 7763 64-Core Processor	8	1	251.7	153,867	462	0.30%	35
Standard_E32-16as_v5	AMD EPYC 7763 64-Core Processor	16	1	251.7	306,591	4,517	1.47%	35
Standard_E48as_v5	AMD EPYC 7763 64-Core Processor	48	1	377.7	892,935	11,974	1.34%	35
Standard_E64as_v5	AMD EPYC 7763 64-Core Processor	64	1	503.7	1,186,352	19,335	1.63%	35
Standard_E64-16as_v5	AMD EPYC 7763 64-Core Processor	16	1	503.7	306,793	869	0.28%	35
Standard_E64-32as_v5	AMD EPYC 7763 64-Core Processor	32	1	503.7	600,716	7,948	1.32%	35
Standard_E96as_v5	AMD EPYC 7763 64-Core Processor	96	2	661.4	1,829,274	23,977	1.31%	35
Standard_E96-24as_v5	AMD EPYC 7763 64-Core Processor	24	1	661.0	448,295	6,035	1.35%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E96-48as_v5	AMD EPYC 7763 64-Core Processor	48	1	661.0	899,801	11,515	1.28%	35
Standard_E96ias_v5	AMD EPYC 7763 64-Core Processor	96	2	661.4	1,836,489	27,407	1.49%	35
Standard_E112ias_v5	AMD EPYC 7763 64-Core Processor	112	2	661.4	2,115,432	31,094	1.47%	35

## EDSv5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	34,923	12	0.04%	35
Standard_E4ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	68,727	286	0.42%	63
Standard_E4-2ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	31.4	34,926	10	0.03%	49
Standard_E8ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	136,905	460	0.34%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8-2ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	62.8	34,902	56	0.16%	42
Standard_E8-4ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	62.8	68,738	252	0.37%	42
Standard_E16ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	271,926	3,097	1.14%	35
Standard_E16-4ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	125.8	68,703	217	0.32%	42
Standard_E16-8ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	125.8	136,880	469	0.34%	49
Standard_E20ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	20	1	157.2	341,049	1,350	0.40%	35
Standard_E32ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	545,162	2,334	0.43%	35
Standard_E32-8ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	251.7	136,995	454	0.33%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32-16ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	251.7	273,226	1,050	0.38%	56
Standard_E48ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	807,259	9,663	1.20%	35
Standard_E64ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	503.9	1,060,197	13,810	1.30%	42
Standard_E64-16ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	2	503.9	273,485	1,014	0.37%	35
Standard_E64-32ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	2	503.9	536,329	9,334	1.74%	42
Standard_E96ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	661.4	1,580,476	17,308	1.10%	35
Standard_E96-24ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	24	2	661.4	397,038	5,317	1.34%	35
Standard_E96-48ds_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	661.4	800,107	8,216	1.03%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E104ids_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	104	2	661.4	1,708,585	21,877	1.28%	42

## EDv5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	34,927	9	0.03%	49
Standard_E4d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	68,699	254	0.37%	42
Standard_E8d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	136,974	509	0.37%	49
Standard_E16d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	273,081	1,049	0.38%	49
Standard_E20d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	20	1	157.2	341,711	1,560	0.46%	42
Standard_E32d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	545,310	2,471	0.45%	56

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E48d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	801,326	9,677	1.21%	42
Standard_E64d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	503.9	1,062,425	13,190	1.24%	49
Standard_E96d_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	661.4	1,584,556	18,706	1.18%	49
Standard_E104id_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	104	2	661.4	1,709,169	17,132	1.00%	35

## ESv5

(03/11/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	31,454	2,465	7.84%	49
Standard_E4s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	65,672	4,558	6.94%	49
Standard_E4-2s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	31.4	32,937	2,633	7.99%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	120,429	6,932	5.76%	49
Standard_E8-2s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	62.8	30,473	1,714	5.62%	35
Standard_E8-4s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	62.8	64,115	4,800	7.49%	42
Standard_E16s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	254,478	19,263	7.57%	56
Standard_E16-4s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	125.8	65,592	5,002	7.63%	49
Standard_E16-8s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	125.8	121,123	7,984	6.59%	35
Standard_E20s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	20	1	157.2	323,191	23,552	7.29%	35
Standard_E32s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	495,411	28,367	5.73%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32-8s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	251.7	123,436	9,678	7.84%	42
Standard_E32-16s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	251.7	232,861	1,867	0.80%	35
Standard_E48s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	699,578	15,956	2.28%	35
Standard_E64s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	503.9	969,853	49,428	5.10%	35
Standard_E64-16s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	2	503.9	244,325	13,378	5.48%	35
Standard_E64-32s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	2	503.9	499,778	35,333	7.07%	35
Standard_E96s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	661.4	1,417,598	86,297	6.09%	35
Standard_E96-24s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	24	2	661.4	368,448	27,903	7.57%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E96-48s_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	661.4	711,917	43,548	6.12%	35
Standard_E104is_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	104	2	661.4	1,718,000	19,930	1.16%	35

## Ev5

(03/14/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	31,147	2,390	7.67%	49
Standard_E4_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	63,068	5,021	7.96%	49
Standard_E8_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	126,346	9,905	7.84%	49
Standard_E16_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	252,327	18,906	7.49%	49
Standard_E20_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	20	1	157.2	307,839	23,401	7.60%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	528,288	18,156	3.44%	42
Standard_E48_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	704,530	25,690	3.65%	35
Standard_E64_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	64	2	503.9	925,660	15,870	1.71%	35
Standard_E96_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	96	2	661.4	1,369,408	18,928	1.38%	35
Standard_E104i_v5	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	104	2	661.4	1,507,969	12,223	0.81%	35

#### Easv4

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2as_v4	AMD EPYC 7452 32-Core Processor	2	1	16.0	38,155	432	1.13%	259
Standard_E4as_v4	AMD EPYC 7452 32-Core Processor	4	1	32.0	71,500	1,260	1.76%	259

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8as_v4	AMD EPYC 7452 32-Core Processor	8	1	64.0	151,201	3,171	2.10%	259
Standard_E16as_v4	AMD EPYC 7452 32-Core Processor	16	2	128.0	293,186	6,371	2.17%	280
Standard_E20as_v4	AMD EPYC 7452 32-Core Processor	20	3	160.0	363,292	8,744	2.41%	280
Standard_E32as_v4	AMD EPYC 7452 32-Core Processor	32	4	256.0	570,199	14,140	2.48%	294
Standard_E48as_v4	AMD EPYC 7452 32-Core Processor	48	6	384.0	841,547	21,858	2.60%	301
Standard_E64as_v4	AMD EPYC 7452 32-Core Processor	64	8	512.0	1,108,151	28,783	2.60%	28
Standard_E96as_v4	AMD EPYC 7452 32-Core Processor	96	12	672.0	1,578,349	27,080	1.72%	42

#### Eav4

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2a_v4	AMD EPYC 7452 32-Core Processor	2	1	16.0	38,035	900	2.37%	196

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	Avg Score	STDDEV	STDDEV%	#RUNS
Standard_E4a_v4	AMD EPYC 7452 32-Core Processor	4	1	32.0	71,345	949	1.33%	196
Standard_E8a_v4	AMD EPYC 7452 32-Core Processor	8	1	64.0	151,511	1,940	1.28%	189
Standard_E16a_v4	AMD EPYC 7452 32-Core Processor	16	2	128.0	292,668	7,220	2.47%	196
Standard_E20a_v4	AMD EPYC 7452 32-Core Processor	20	3	160.0	361,845	9,332	2.58%	196
Standard_E32a_v4	AMD EPYC 7452 32-Core Processor	32	4	256.0	570,615	14,419	2.53%	196
Standard_E48a_v4	AMD EPYC 7452 32-Core Processor	48	6	384.0	841,419	26,181	3.11%	196
Standard_E64a_v4	AMD EPYC 7452 32-Core Processor	64	8	512.0	1,116,891	28,545	2.56%	14
Standard_E96a_v4	AMD EPYC 7452 32-Core Processor	96	12	672.0	1,592,228	32,515	2.04%	14

EDSv4

(09/22/2020 PBIID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
---------	-----	-------	------------	--------------	-----------	--------	---------	-------

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	35,975	673	1.87%	189
Standard_E4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	65,819	523	0.79%	182
Standard_E4-2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	32.0	35,623	312	0.88%	189
Standard_E8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	135,576	723	0.53%	189
Standard_E8-2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	64.0	35,831	529	1.48%	189
Standard_E8-4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	64.0	66,007	778	1.18%	189
Standard_E16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	271,138	1,608	0.59%	189
Standard_E16-4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	128.0	66,461	1,250	1.88%	189

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16-8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	128.0	135,960	1,073	0.79%	189
Standard_E20ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	339,000	1,496	0.44%	189
Standard_E32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	538,949	6,669	1.24%	168
Standard_E32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	256.0	527,258	11,587	2.20%	21
Standard_E32-8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	256.0	136,425	1,166	0.85%	147
Standard_E32-8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	2	256.0	136,743	2,124	1.55%	42
Standard_E32-16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	256.0	271,264	1,345	0.50%	154
Standard_E32-16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	256.0	269,101	4,332	1.61%	28

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	775,434	10,356	1.34%	189
Standard_E64ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,027,169	12,890	1.25%	182
Standard_E64-16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	504.0	266,956	4,181	1.57%	189
Standard_E64-32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	504.0	521,183	6,655	1.28%	182

## EDv4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	35,901	490	1.36%	189
Standard_E4d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	65,859	786	1.19%	189
Standard_E8d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	135,986	1,194	0.88%	189

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	271,196	1,448	0.53%	182
Standard_E20d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	338,449	2,948	0.87%	189
Standard_E32d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	539,487	5,562	1.03%	161
Standard_E32d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	256.0	529,841	13,344	2.52%	28
Standard_E48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	772,684	11,745	1.52%	189
Standard_E64d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,025,563	13,902	1.36%	189

#### Esv4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	35,750	491	1.37%	175

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	65,919	696	1.06%	182
Standard_E4-2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	32.0	35,875	598	1.67%	182
Standard_E8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	136,061	1,199	0.88%	182
Standard_E8-2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	64.0	35,762	404	1.13%	182
Standard_E8-4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	64.0	65,865	530	0.81%	182
Standard_E16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	270,956	1,486	0.55%	182
Standard_E16-4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	128.0	66,290	1,010	1.52%	182
Standard_E16-8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	128.0	135,990	1,139	0.84%	182

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E20s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	339,149	1,689	0.50%	182
Standard_E32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	540,883	3,428	0.63%	154
Standard_E32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	256.0	533,105	10,700	2.01%	28
Standard_E32-8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	256.0	136,061	1,055	0.78%	140
Standard_E32-8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	2	256.0	136,911	3,347	2.44%	35
Standard_E32-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	256.0	271,570	1,334	0.49%	147
Standard_E32-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	256.0	264,434	5,182	1.96%	35
Standard_E48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	774,071	9,926	1.28%	182

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E64s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,024,690	13,752	1.34%	182
Standard_E64-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	504.0	266,980	3,951	1.48%	182
Standard_E64-32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	504.0	521,342	7,052	1.35%	182

#### Ev4

(09/22/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	35,781	422	1.18%	182
Standard_E4_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	65,742	489	0.74%	175
Standard_E8_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	135,668	847	0.62%	182
Standard_E16_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	270,553	2,565	0.95%	182

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E20_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	338,166	4,445	1.31%	175
Standard_E32_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	541,040	2,289	0.42%	154
Standard_E32_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	256.0	520,279	6,500	1.25%	28
Standard_E48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	774,116	10,210	1.32%	175
Standard_E64_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,025,603	13,715	1.34%	182

### Esv3 - Memory Optimized + Premium Storage

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	16.0	27,008	2,136	7.91%	105
Standard_E2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	16.0	26,729	987	3.69%	98

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	31,341	14	0.04%	21
Standard_E4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	32.0	51,647	3,354	6.49%	105
Standard_E4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	32.0	49,459	2,426	4.91%	98
Standard_E4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	57,986	283	0.49%	35
Standard_E4-2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	32.0	26,468	1,997	7.54%	105
Standard_E4-2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	32.0	26,457	817	3.09%	84
Standard_E4-2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	32.0	31,350	15	0.05%	42
Standard_E8s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	64.0	101,750	7,494	7.36%	126

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	64.0	101,526	1,976	1.95%	77
Standard_E8s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	120,073	826	0.69%	35
Standard_E8-2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	64.0	26,318	1,665	6.33%	140
Standard_E8-2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	64.0	27,219	1,167	4.29%	56
Standard_E8-2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	64.0	31,322	38	0.12%	42
Standard_E8-4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	64.0	52,719	3,706	7.03%	119
Standard_E8-4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	64.0	49,518	2,675	5.40%	77
Standard_E8-4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	64.0	57,859	378	0.65%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	128.0	195,755	5,337	2.73%	126
Standard_E16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	128.0	199,681	2,067	1.04%	77
Standard_E16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	240,250	984	0.41%	28
Standard_E16-4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	128.0	54,920	3,833	6.98%	140
Standard_E16-4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	128.0	50,869	3,400	6.68%	77
Standard_E16-4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	128.0	58,019	307	0.53%	21
Standard_E16-8s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	128.0	103,358	6,272	6.07%	133
Standard_E16-8s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	128.0	102,945	2,982	2.90%	63

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16-8s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	128.0	120,189	963	0.80%	35
Standard_E20s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	160.0	245,271	1,479	0.60%	112
Standard_E20s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	160.0	249,245	2,057	0.83%	98
Standard_E20s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	300,197	1,881	0.63%	28
Standard_E32s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	2	256.0	379,015	6,761	1.78%	126
Standard_E32s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	256.0	397,086	2,450	0.62%	84
Standard_E32s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	477,740	4,873	1.02%	28
Standard_E32-8s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	2	256.0	105,799	5,853	5.53%	98

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32-8s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	256.0	103,535	2,602	2.51%	105
Standard_E32-8s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	256.0	120,109	593	0.49%	28
Standard_E32-16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	2	256.0	199,765	7,562	3.79%	119
Standard_E32-16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	256.0	202,033	3,104	1.54%	77
Standard_E32-16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	256.0	240,249	1,126	0.47%	42
Standard_E48s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	48	2	384.0	570,727	4,290	0.75%	98
Standard_E48s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	2	384.0	578,285	5,864	1.01%	105
Standard_E48s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	687,857	6,709	0.98%	35

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E64s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	64	2	432.0	763,127	5,541	0.73%	42
Standard_E64s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	64	2	432.0	770,574	7,440	0.97%	161
Standard_E64s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	432.0	913,390	9,287	1.02%	21
Standard_E64-16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	2	432.0	234,395	2,364	1.01%	42
Standard_E64-16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	2	432.0	211,781	7,841	3.70%	161
Standard_E64-16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	432.0	232,571	3,046	1.31%	28
Standard_E64-32s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	2	432.0	411,194	4,489	1.09%	56
Standard_E64-32s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	2	432.0	395,227	5,245	1.33%	161

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E64-32s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	432.0	464,744	6,855	1.48%	21

### Ev3 - Memory Optimized

(09/23/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	16.0	27,615	2,226	8.06%	133
Standard_E2_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	16.0	27,471	1,897	6.90%	84
Standard_E4_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	32.0	51,755	3,604	6.96%	175
Standard_E4_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	32.0	48,159	1,286	2.67%	63
Standard_E8_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	64.0	100,794	5,678	5.63%	154
Standard_E8_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	64.0	100,458	1,381	1.38%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	128.0	196,170	4,590	2.34%	189
Standard_E16_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	128.0	199,986	1,534	0.77%	42
Standard_E16_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	240,246	680	0.28%	7
Standard_E20_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	160.0	243,816	2,644	1.08%	196
Standard_E20_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	160.0	245,193	6,282	2.56%	35
Standard_E20_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	300,653	666	0.22%	7
Standard_E32_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	2	256.0	377,052	7,434	1.97%	147
Standard_E32_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	256.0	393,569	5,278	1.34%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	479,255	2,237	0.47%	14
Standard_E48_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	48	2	384.0	569,428	5,166	0.91%	147
Standard_E48_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	48	2	384.0	577,766	6,460	1.12%	77
Standard_E48_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	687,699	7,983	1.16%	14
Standard_E64_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	64	2	432.0	761,587	5,928	0.78%	63
Standard_E64_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	64	2	432.0	765,301	11,204	1.46%	161
Standard_E64_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	432.0	913,332	12,240	1.34%	14

#### DSv2 - Memory Optimized + Premium Storage

(09/30/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS11_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	14.0	37,489	1,314	3.50%	49
Standard_DS11_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	14.0	40,138	3,286	8.19%	70
Standard_DS11_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	14.0	37,889	1,096	2.89%	84
Standard_DS11_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	14.0	44,107	406	0.92%	28
Standard_DS11-1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	14.0	19,615	846	4.31%	42
Standard_DS11-1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	14.0	20,835	1,937	9.30%	42
Standard_DS11-1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	14.0	18,765	767	4.09%	126
Standard_DS11-1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	14.0	23,145	1,425	6.16%	21

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS12_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	28.0	73,222	1,506	2.06%	42
Standard_DS12_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	28.0	76,452	5,858	7.66%	77
Standard_DS12_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	28.0	73,140	3,323	4.54%	84
Standard_DS12_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	28.0	86,916	1,044	1.20%	28
Standard_DS12-1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	28.0	19,387	899	4.64%	56
Standard_DS12-1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	28.0	22,048	1,729	7.84%	77
Standard_DS12-1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	28.0	19,102	943	4.94%	70
Standard_DS12-1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	28.0	22,205	32	0.14%	28

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS12-2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	28.0	38,766	1,060	2.74%	56
Standard_DS12-2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	28.0	41,447	3,954	9.54%	91
Standard_DS12-2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	28.0	37,639	1,579	4.19%	70
Standard_DS12-2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	28.0	44,226	810	1.83%	14
Standard_DS13_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	56.0	147,234	1,202	0.82%	56
Standard_DS13_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	56.0	147,218	1,636	1.11%	70
Standard_DS13_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	56.0	144,612	1,582	1.09%	84
Standard_DS13_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	56.0	172,617	755	0.44%	21

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS13-2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	56.0	38,688	816	2.11%	56
Standard_DS13-2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	56.0	39,898	2,493	6.25%	84
Standard_DS13-2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	56.0	38,861	2,788	7.17%	91
Standard_DS13-4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	56.0	73,105	1,727	2.36%	70
Standard_DS13-4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	56.0	77,848	4,307	5.53%	84
Standard_DS13-4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	56.0	74,935	4,384	5.85%	63
Standard_DS13-4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	56.0	86,402	529	0.61%	14
Standard_DS14_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	112.0	284,796	4,889	1.72%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS14_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	112.0	298,261	1,351	0.45%	63
Standard_DS14_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	112.0	286,993	3,418	1.19%	98
Standard_DS14_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	112.0	345,521	1,549	0.45%	21
Standard_DS14-4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	2	112.0	71,287	2,692	3.78%	42
Standard_DS14-4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	112.0	81,710	6,156	7.53%	56
Standard_DS14-4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	112.0	75,101	2,765	3.68%	112
Standard_DS14-4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	112.0	86,598	517	0.60%	21
Standard_DS14-8_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	2	112.0	143,882	4,104	2.85%	49

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS14-8_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	112.0	150,389	4,548	3.02%	70
Standard_DS14-8_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	112.0	145,955	2,224	1.52%	105
Standard_DS14-8_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	112.0	171,910	754	0.44%	7
Standard_DS15_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	140.0	355,354	6,693	1.88%	49
Standard_DS15_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	140.0	421,202	56,292	13.36%	154

## Dv2 - Memory Optimized Compute

(09/30/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D11_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	14.0	37,655	1,925	5.11%	70
Standard_D11_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	14.0	39,953	3,019	7.56%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D11_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	14.0	37,979	1,830	4.82%	56
Standard_D12_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	28.0	71,521	4,218	5.90%	77
Standard_D12_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	28.0	77,439	5,095	6.58%	70
Standard_D12_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	28.0	73,464	1,263	1.72%	56
Standard_D12_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	28.0	87,100	434	0.50%	7
Standard_D13_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	56.0	141,552	16,057	11.34%	70
Standard_D13_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	56.0	148,352	2,754	1.86%	70
Standard_D13_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	56.0	141,946	3,604	2.54%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D13_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	56.0	167,641	1,680	1.00%	7
Standard_D14_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	112.0	283,793	5,207	1.83%	77
Standard_D14_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	112.0	297,163	1,817	0.61%	63
Standard_D14_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	112.0	286,817	4,109	1.43%	49
Standard_D15_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	140.0	354,279	6,021	1.70%	35
Standard_D15_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	140.0	418,475	58,044	13.87%	154

## Msv2 High Memory

(03/17/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M208s_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	208	4	2,850.0	3,020,762	55,134	1.83%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M208ms_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	208	4	5,700.0	3,009,120	58,843	1.96%	42
Standard_M208ms_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	208	8	5,700.0	3,093,184	33,253	1.08%	42
Standard_M416s_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	416	8	5,700.0	5,959,252	93,933	1.58%	84
Standard_M416ms_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	416	8	11,400.0	5,910,261	101,190	1.71%	84
Standard_M416-208s_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	208	8	5,700.0	3,085,232	36,568	1.19%	70
Standard_M416-208ms_v2	Intel(R) Xeon(R) Platinum 8180M CPU @ 2.50GHz	208	8	11,400.0	3,064,892	40,531	1.32%	77

## Msv2 Medium Memory

(03/16/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M32ms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	2	861.2	501,859	6,988	1.39%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M32dms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	2	861.2	507,318	12,103	2.39%	35
Standard_M64s_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,007.9	977,226	15,256	1.56%	35
Standard_M64ds_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,007.9	980,928	16,781	1.71%	35
Standard_M64ms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	4	1,763.9	980,511	16,384	1.67%	42
Standard_M64dms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	4	1,763.9	979,367	13,927	1.42%	35
Standard_M128s_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	2,015.9	1,905,457	53,830	2.83%	35
Standard_M128ds_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	2,015.9	1,925,932	29,177	1.51%	35
Standard_M128ms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	3,831.1	1,907,485	30,641	1.61%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M128dm_s_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	3,831.1	1,907,006	25,958	1.36%	35
Standard_M192ms_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	192	4	4,031.9	2,794,826	44,549	1.59%	84
Standard_M192is_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	192	4	2,015.9	2,805,023	39,961	1.42%	35
Standard_M192ims_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	192	4	4,031.9	2,797,557	39,161	1.40%	35
Standard_M192ids_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	192	4	2,015.9	2,818,523	48,260	1.71%	35
Standard_M192idm_s_v2	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	192	4	4,031.9	2,813,406	47,652	1.69%	35

### M-series Medium Memory

(09/29/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M64	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	2	1,024.0	821,678	11,118	1.35%	77

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M64	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,024.0	754,180	5,686	0.75%	7
Standard_M64m	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	2	1,792.0	817,400	9,397	1.15%	56
Standard_M64m	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,792.0	754,929	10,566	1.40%	28
Standard_M128	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	128	4	2,048.0	1,643,769	17,798	1.08%	70
Standard_M128	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	2,048.0	1,475,843	16,197	1.10%	14
Standard_M128m	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	128	4	3,892.0	1,631,313	18,162	1.11%	70
Standard_M128m	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	3,892.0	1,471,781	17,646	1.20%	14
Standard_M8ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	8	1	218.8	110,370	366	0.33%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M8-2ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	2	1	218.8	28,041	35	0.12%	84
Standard_M8-4ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	4	1	218.8	56,231	348	0.62%	77
Standard_M8-4ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	4	1	218.8	48,391	280	0.58%	7
Standard_M16ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	16	1	437.5	220,331	473	0.21%	70
Standard_M16ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	16	1	437.5	200,561	877	0.44%	14
Standard_M16-4ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	4	1	437.5	56,218	304	0.54%	70
Standard_M16-4ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	4	1	437.5	48,401	205	0.42%	14
Standard_M16-8ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	8	1	437.5	110,426	383	0.35%	56

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M16-8ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	8	1	437.5	100,377	442	0.44%	28
Standard_M32ls	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	1	256.0	433,173	4,290	0.99%	70
Standard_M32ls	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	1	256.0	400,287	1,610	0.40%	14
Standard_M32ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	1	875.0	434,866	2,560	0.59%	70
Standard_M32ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	1	875.0	400,583	1,596	0.40%	14
Standard_M32-8ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	8	1	875.0	110,408	346	0.31%	56
Standard_M32-8ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	8	1	875.0	100,521	643	0.64%	28
Standard_M32-16ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	16	1	875.0	220,495	569	0.26%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M32-16ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	16	1	875.0	200,494	723	0.36%	14
Standard_M32ts	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	1	192.0	435,785	2,164	0.50%	77
Standard_M32ts	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	1	192.0	399,771	1,291	0.32%	7
Standard_M64s	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	2	1,024.0	822,245	7,256	0.88%	70
Standard_M64s	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,024.0	750,702	9,866	1.31%	14
Standard_M64ls	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	2	512.0	820,406	13,530	1.65%	77
Standard_M64ls	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	512.0	759,753	7,639	1.01%	7
Standard_M64ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	2	1,792.0	823,449	7,565	0.92%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M64ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	2	1,792.0	755,541	9,337	1.24%	14
Standard_M64-16ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	16	2	1,792.0	210,947	2,766	1.31%	77
Standard_M64-16ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	16	2	1,792.0	191,761	3,503	1.83%	7
Standard_M64-32ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	2	1,792.0	416,909	3,769	0.90%	63
Standard_M64-32ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	3	1,792.0	417,771	4,820	1.15%	7
Standard_M64-32ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	2	1,792.0	382,773	3,547	0.93%	14
Standard_M128s	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	128	4	2,048.0	1,641,176	16,698	1.02%	70
Standard_M128s	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	2,048.0	1,477,253	15,626	1.06%	14

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_M128ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	128	4	3,892.0	1,645,242	17,679	1.07%	63
Standard_M128ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	128	4	3,892.0	1,479,298	16,978	1.15%	21
Standard_M128-32ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	32	4	3,892.0	414,541	4,739	1.14%	63
Standard_M128-32ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	32	4	3,892.0	380,611	5,653	1.49%	21
Standard_M128-64ms	Intel(R) Xeon(R) CPU E7-8890 v3 @ 2.50GHz	64	4	3,892.0	817,496	8,437	1.03%	56
Standard_M128-64ms	Intel(R) Xeon(R) Platinum 8280M CPU @ 2.70GHz	64	4	3,892.0	749,800	8,506	1.13%	28

## Storage optimized

### Lsv2 - Storage Optimized

(10/13/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_L8s_v2	AMD EPYC 7551 32-Core Processor	8	1	64.0	102,237	1,320	1.29%	77

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_L16s_v2	AMD EPYC 7551 32-Core Processor	16	2	128.0	198,472	3,734	1.88%	70
Standard_L32s_v2	AMD EPYC 7551 32-Core Processor	32	4	256.0	390,015	10,126	2.60%	77
Standard_L48s_v2	AMD EPYC 7551 32-Core Processor	48	6	384.0	583,388	15,479	2.65%	77
Standard_L64s_v2	AMD EPYC 7551 32-Core Processor	64	8	512.0	774,827	19,205	2.48%	77
Standard_L80s_v2	AMD EPYC 7551 32-Core Processor	80	10	640.0	966,682	22,811	2.36%	77

## Confidential Compute

### DCSv3

(04/05/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC1s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	1	1	7.8	29,127	72	0.25%	56
Standard_DC2s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	55,785	534	0.96%	56

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC4s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	111,636	827	0.74%	63
Standard_DC8s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	220,443	2,083	0.94%	56
Standard_DC16s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	439,985	4,224	0.96%	56
Standard_DC24s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	24	1	188.7	659,602	5,809	0.88%	42
Standard_DC32s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	850,747	8,658	1.02%	63
Standard_DC48s_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	1,221,232	25,082	2.05%	63

### DCDSv3

(04/05/2022 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC1ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	1	1	7.8	29,130	66	0.23%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC2ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	2	1	15.6	55,911	207	0.37%	56
Standard_DC4ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	4	1	31.4	111,549	745	0.67%	56
Standard_DC8ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	8	1	62.8	220,223	2,067	0.94%	42
Standard_DC16ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	16	1	125.8	440,580	3,915	0.89%	63
Standard_DC24ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	24	1	188.7	656,455	9,830	1.50%	56
Standard_DC32ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	32	1	251.7	848,418	18,613	2.19%	35
Standard_DC48ds_v3	Intel(R) Xeon(R) Platinum 8370C CPU @ 2.80GHz	48	2	377.9	1,228,361	21,388	1.74%	35

## DCsv2

(10/08/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC1s_v2	Intel(R) Xeon(R) E-2288G CPU @ 3.70GHz	1	1	4.0	34,418	162	0.47%	77
Standard_DC2s_v2	Intel(R) Xeon(R) E-2288G CPU @ 3.70GHz	2	1	8.0	68,562	758	1.11%	77
Standard_DC4s_v2	Intel(R) Xeon(R) E-2288G CPU @ 3.70GHz	4	1	16.0	133,836	1,964	1.47%	77

## DCv2

(10/13/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC8_v2	Intel(R) Xeon(R) E-2288G CPU @ 3.70GHz	8	1	32.0	252,047	3,051	1.21%	77

## DCS

(10/01/2020 PBID:7668456)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DC2s	Intel(R) Xeon(R) E-2176G CPU @ 3.70GHz	2	1	8.0	63,426	391	0.62%	28
Standard_DC4s	Intel(R) Xeon(R) E-2176G CPU @ 3.70GHz	4	1	16.0	124,015	410	0.33%	21

## About CoreMark

CoreMark is a benchmark that tests the functionality of a microcontroller (MCU) or central processing unit (CPU). CoreMark is not system dependent, so it functions the same regardless of the platform (e.g. big or little endian, high-end or low-end processor).

Linux numbers were computed by running CoreMark on Ubuntu 18.04. CoreMark was configured with the

number of threads set to the number of virtual CPUs, and concurrency set to `PThreads`. The target number of iterations was adjusted based on expected performance to provide a runtime of at least 20 seconds (typically much longer). The final score represents the number of iterations completed divided by the number of seconds it took to run the test. Each test was run at least seven times on each VM. Test run dates shown above. Tests run on multiple VMs across Azure public regions the VM was supported in on the date run.

## Running Coremark on Azure VMs

### Download:

CoreMark is an open source tool that can be downloaded from [GitHub](#).

### Building and Running:

To build and run the benchmark, type:

```
> make
```

Full results are available in the files `run1.log` and `run2.log`. `run1.log` contains CoreMark results. These are the benchmark results with performance parameters. `run2.log` contains benchmark results with validation parameters.

### Run Time:

By default, the benchmark will run between 10-100 seconds. To override, use `ITERATIONS=N`

```
% make ITERATIONS=10
```

above flag will run the benchmark for 10 iterations. **Results are only valid for reporting if the benchmark ran for at least 10 seconds!**

### Parallel Execution:

Use `XCFLAGS=-DMULTITHREAD=N` where N is number of threads to run in parallel. Several implementations are available to execute in multiple contexts.

```
% make XCFLAGS="-DMULTITHREAD=4 -DUSE_PTHREAD"
```

The above will compile the benchmark for execution on 4 cores.

### Recommendations for best results

- The benchmark needs to run for at least 10 seconds, probably longer on larger systems.
- All source files must be compiled with same flags.
- Do not change source files other than `core_portme*` (use `make check` to validate)
- Multiple runs are suggested for best results.

## Coverage

Older end-of-life series may not be shown. N series not shown as they are GPU centric and Coremark doesn't measure GPU performance. Newer series may not have been benchmarked yet. Previous versions of this document cited benchmark runs from Ubuntu 16.04 which resulted in slightly lower performance than the current benchmarks running on Ubuntu 18.04.

## GPU Series

Performance of GPU based VM series is best understood by using GPU appropriate benchmarks and running at the scale required for your workloads. Azure ranks among the best there:

- Top 10 Supercomputer: [November 2021 | TOP500](#) (Azure powered #10: Voyager-EUS2)

- Machine Learning: MLCommons Training: [v1.1 Results](#) | [MLCommons](#) (2 highest at scale and largest in the cloud)

## Next steps

- For storage capacities, disk details, and additional considerations for choosing among VM sizes, see [Sizes for virtual machines](#).

# Compute benchmark scores for Windows VMs

9/21/2022 • 40 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

The following CoreMark benchmark scores show compute performance for select Azure VMs running Windows Server 2019. Compute benchmark scores are also available for [Linux VMs](#).

TYPE	FAMILIES
Compute optimized	Fsv2, Fs, F, GS, G
General purpose	B, Dasv4, Dav4, Ddsv4, Ddv4, Dsv4, Dv4, Dsv3, Dv3, DSv2, Dv2, Av2
High performance compute	HB, HC
Memory optimized	Easv4, Eav4, Edsv4, Edv4, Esv4, Ev4, Esv3, Ev3, DSv2, Dv2
Storage optimized	Lsv2, Ls

## Compute optimized

### Fsv2 - Compute + Premium Storage

(03/29/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F2s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	2	1	4.0	34,903	1,101	3.15%	112
Standard_F2s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	2	1	4.0	34,738	1,331	3.83%	224

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F4s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	4	1	8.0	66,903	1,047	1.57%	182
Standard_F8s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	8	1	16.0	131,477	2,180	1.66%	140
Standard_F8s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	8	1	16.0	132,533	1,732	1.31%	210
Standard_F16s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	16	1	32.0	260,760	3,629	1.39%	112
Standard_F16s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	16	1	32.0	265,158	2,185	0.82%	182
Standard_F32s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	32	1	64.0	525,608	6,270	1.19%	98
Standard_F32s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	32	1	64.0	530,137	6,085	1.15%	140
Standard_F48s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	48	2	96.0	769,768	7,567	0.98%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F48s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	48	1	96.0	742,828	17,316	2.33%	112
Standard_F64s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	64	2	128.0	1,030,552	8,106	0.79%	70
Standard_F64s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	64	2	128.0	1,028,052	9,373	0.91%	168
Standard_F72s_v2	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70 GHz	72	2	144.0	N/A	-	-	-
Standard_F72s_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	72	2	144.0	N/A	-	-	-

### Fs - Compute Optimized + Premium Storage

(04/28/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F1s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	1	1	2.0	16,445	825	5.02%	42
Standard_F1s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	1	1	2.0	17,614	2,873	16.31%	210

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F1s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	1	1	2.0	16,053	1,802	11.22%	70
Standard_F1s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	1	1	2.0	20,007	1,684	8.42%	28
Standard_F2s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	2	1	4.0	33,451	3,424	10.24%	70
Standard_F2s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	2	1	4.0	33,626	2,990	8.89%	154
Standard_F2s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	2	1	4.0	34,386	3,851	11.20%	98
Standard_F2s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	2	1	4.0	36,826	344	0.94%	28
Standard_F4s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	4	1	8.0	67,351	4,407	6.54%	42
Standard_F4s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	4	1	8.0	67,009	4,637	6.92%	196

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F4s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	4	1	8.0	63,668	3,375	5.30%	84
Standard_F4s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	4	1	8.0	79,153	15,034	18.99%	28
Standard_F8s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	8	1	16.0	128,232	1,272	0.99%	42
Standard_F8s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	8	1	16.0	127,871	5,109	4.00%	154
Standard_F8s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	8	1	16.0	122,811	5,481	4.46%	126
Standard_F8s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	8	1	16.0	154,842	10,354	6.69%	28
Standard_F16s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	16	2	32.0	260,883	15,853	6.08%	42
Standard_F16s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	16	1	32.0	255,762	4,966	1.94%	182

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F16s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	16	1	32.0	248,884	11,035	4.43%	70
Standard_F16s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	16	1	32.0	310,303	21,942	7.07%	28

## F - Compute Optimized

(04/28/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F1	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	1	1	2.0	17,356	1,151	6.63%	112
Standard_F1	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	1	1	2.0	16,508	1,740	10.54%	154
Standard_F1	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	1	1	2.0	16,076	2,065	12.84%	70
Standard_F1	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	1	1	2.0	20,074	1,612	8.03%	14
Standard_F2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	2	1	4.0	32,770	1,915	5.84%	126

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	2	1	4.0	33,081	2,242	6.78%	126
Standard_F2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	2	1	4.0	33,310	2,532	7.60%	84
Standard_F2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	2	1	4.0	40,746	2,027	4.98%	14
Standard_F4	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	4	1	8.0	65,694	3,512	5.35%	126
Standard_F4	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	4	1	8.0	65,054	3,457	5.31%	154
Standard_F4	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	4	1	8.0	61,607	3,662	5.94%	56
Standard_F4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	4	1	8.0	76,884	1,763	2.29%	14
Standard_F8	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	8	1	16.0	130,415	5,353	4.10%	98

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_F8	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	8	1	16.0	126,139	2,917	2.31%	126
Standard_F8	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	8	1	16.0	122,443	4,391	3.59%	98
Standard_F8	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	8	1	16.0	144,696	2,172	1.50%	14
Standard_F16	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	16	2	32.0	253,473	8,597	3.39%	140
Standard_F16	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	16	1	32.0	257,457	7,596	2.95%	126
Standard_F16	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	16	1	32.0	244,559	8,036	3.29%	70
Standard_F16	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	16	1	32.0	283,565	8,683	3.06%	14

#### GS - Compute Optimized + Premium Storage

(05/27/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_GS1	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	2	1	28.0	35,593	2,888	8.11%	252
Standard_GS2	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	4	1	56.0	72,188	5,949	8.24%	252
Standard_GS3	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	8	1	112.0	132,665	6,910	5.21%	238
Standard_GS4	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	16	1	224.0	261,542	3,722	1.42%	252
Standard_GS4-4	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	4	1	224.0	79,352	4,935	6.22%	224
Standard_GS4-8	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	8	1	224.0	137,774	6,887	5.00%	238
Standard_GS5	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	32	2	448.0	507,026	6,895	1.36%	252
Standard_GS5-8	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	8	2	448.0	157,541	3,151	2.00%	238

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_GS5-16	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	16	2	448.0	278,656	5,235	1.88%	224

## G - Compute Optimized

(05/27/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_G1	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	2	1	28.0	36,386	4,100	11.27%	252
Standard_G2	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	4	1	56.0	72,484	5,563	7.67%	252
Standard_G3	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	8	1	112.0	136,618	5,714	4.18%	252
Standard_G4	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	16	1	224.0	261,708	3,426	1.31%	238
Standard_G5	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00 GHz	32	2	448.0	507,423	7,261	1.43%	252

## General purpose

### B - Burstable

(04/12/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B1ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	1	1	2.0	18,093	679	3.75%	42
Standard_B1ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	1	1	2.0	18,197	1,341	7.37%	168
Standard_B1ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	1	1	2.0	17,975	920	5.12%	112
Standard_B1ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	1	1	2.0	20,176	1,568	7.77%	28
Standard_B2s	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	2	1	4.0	35,546	660	1.86%	42
Standard_B2s	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	2	1	4.0	36,569	2,172	5.94%	154
Standard_B2s	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	2	1	4.0	36,136	924	2.56%	140
Standard_B2s	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	2	1	4.0	42,546	834	1.96%	14

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B2hms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	2	1	8.0	36,949	1,494	4.04%	28
Standard_B2hms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	2	1	8.0	36,512	2,537	6.95%	70
Standard_B2hms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	2	1	8.0	36,389	990	2.72%	56
Standard_B2ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	2	1	8.0	35,758	1,028	2.88%	42
Standard_B2ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	2	1	8.0	36,028	1,605	4.45%	182
Standard_B2ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	2	1	8.0	36,122	2,128	5.89%	112
Standard_B2ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	2	1	8.0	42,525	672	1.58%	14
Standard_B4hms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	4	1	16.0	71,028	879	1.24%	28

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B4hms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	4	1	16.0	73,126	2,954	4.04%	56
Standard_B4hms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	4	1	16.0	68,451	1,571	2.29%	56
Standard_B4hms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	4	1	16.0	83,525	563	0.67%	14
Standard_B4ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	4	1	16.0	70,831	1,135	1.60%	28
Standard_B4ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	4	1	16.0	70,987	2,287	3.22%	168
Standard_B4ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	4	1	16.0	68,796	1,897	2.76%	84
Standard_B4ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	4	1	16.0	81,712	4,042	4.95%	70
Standard_B8ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	8	1	32.0	141,620	2,256	1.59%	42

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B8ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	8	1	32.0	139,090	3,229	2.32%	182
Standard_B8ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	8	1	32.0	135,510	2,653	1.96%	112
Standard_B8ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	8	1	32.0	164,510	2,254	1.37%	14
Standard_B12ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	12	1	48.0	206,957	5,240	2.53%	56
Standard_B12ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	12	1	48.0	211,461	4,115	1.95%	154
Standard_B12ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	12	1	48.0	200,729	3,475	1.73%	140
Standard_B16ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	16	2	64.0	273,257	3,862	1.41%	42
Standard_B16ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	16	1	64.0	282,187	5,030	1.78%	154

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_B16ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	16	1	64.0	265,834	5,545	2.09%	112
Standard_B16ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	16	1	64.0	331,694	3,537	1.07%	28
Standard_B20ms	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40 GHz	20	2	80.0	334,369	8,555	2.56%	42
Standard_B20ms	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz	20	1	80.0	345,686	6,702	1.94%	154
Standard_B20ms	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60 GHz	20	1	80.0	328,900	7,625	2.32%	126
Standard_B20ms	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60 GHz	20	1	80.0	409,515	4,792	1.17%	14

#### Dasv4

(03/25/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2as_v4	AMD EPYC 7452 32-Core Processor	2	1	8.0	37,986	1,199	3.16%	238

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4as_v4	AMD EPYC 7452 32-Core Processor	4	1	16.0	75,828	1,343	1.77%	196
Standard_D8as_v4	AMD EPYC 7452 32-Core Processor	8	1	32.0	150,134	2,511	1.67%	210
Standard_D16as_v4	AMD EPYC 7452 32-Core Processor	16	2	64.0	286,789	5,984	2.09%	224
Standard_D32as_v4	AMD EPYC 7452 32-Core Processor	32	4	128.0	566,270	8,484	1.50%	140
Standard_D48as_v4	AMD EPYC 7452 32-Core Processor	48	6	192.0	829,547	15,679	1.89%	126
Standard_D64as_v4	AMD EPYC 7452 32-Core Processor	64	8	256.0	1,088,030	16,708	1.54%	28
Standard_D96as_v4	AMD EPYC 7452 32-Core Processor	96	12	384.0	N/A	-	-	-

#### Dav4

(03/25/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2a_v4	AMD EPYC 7452 32-Core Processor	2	1	8.0	38,028	995	2.62%	238

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4a_v4	AMD EPYC 7452 32-Core Processor	4	1	16.0	75,058	1,874	2.50%	238
Standard_D8a_v4	AMD EPYC 7452 32-Core Processor	8	1	32.0	149,706	2,520	1.68%	168
Standard_D16a_v4	AMD EPYC 7452 32-Core Processor	16	2	64.0	287,479	4,907	1.71%	238
Standard_D32a_v4	AMD EPYC 7452 32-Core Processor	32	4	128.0	567,019	11,019	1.94%	210
Standard_D48a_v4	AMD EPYC 7452 32-Core Processor	48	6	192.0	835,617	13,097	1.57%	140
Standard_D64a_v4	AMD EPYC 7452 32-Core Processor	64	8	256.0	1,099,165	21,962	2.00%	252
Standard_D96a_v4	AMD EPYC 7452 32-Core Processor	96	12	384.0	N/A	-	-	-

## DDSV4

(03/26/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	34,621	1,588	4.59%	336

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	66,583	2,327	3.49%	336
Standard_D8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	131,888	3,913	2.97%	336
Standard_D16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	262,436	9,177	3.50%	336
Standard_D32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	531,747	5,956	1.12%	322
Standard_D48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	750,843	15,060	2.01%	420
Standard_D48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	753,948	31,559	4.19%	252

#### DDv4

(03/26/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	34,704	1,455	4.19%	336

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	66,629	2,005	3.01%	336
Standard_D8d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	131,953	3,911	2.96%	336
Standard_D16d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	263,568	7,317	2.78%	336
Standard_D32d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	527,571	11,076	2.10%	336
Standard_D48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	742,908	19,323	2.60%	378
Standard_D48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	759,921	18,783	2.47%	280

#### Dsv4

(03/24/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	31,643	3,054	9.65%	406

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	60,878	4,594	7.55%	392
Standard_D8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	119,076	7,683	6.45%	406
Standard_D16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	242,206	16,772	6.92%	406
Standard_D32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	483,021	28,105	5.82%	392
Standard_D48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	694,366	33,144	4.77%	280
Standard_D48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	705,192	24,651	3.50%	126
Standard_D64s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,023,014	17,746	1.73%	364

#### Dv4

(03/25/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	31,469	2,948	9.37%	406
Standard_D4_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	61,806	4,467	7.23%	406
Standard_D8_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	120,421	8,407	6.98%	392
Standard_D16_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	245,522	17,151	6.99%	812
Standard_D32_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	128.0	487,165	28,119	5.77%	378
Standard_D48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	1	192.0	688,018	24,945	3.63%	252
Standard_D48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	192.0	696,691	30,283	4.35%	112
Standard_D64_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	256.0	1,018,300	23,085	2.27%	392

#### DSv3 - General Compute + Premium Storage

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	8.0	23,534	724	3.08%	42
Standard_D2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	8.0	24,742	2,045	8.27%	112
Standard_D2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	8.0	24,822	3,702	14.91%	126
Standard_D2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	8.0	30,392	1,514	4.98%	28
Standard_D4s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	16.0	44,404	537	1.21%	28
Standard_D4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	16.0	45,725	4,388	9.60%	154
Standard_D4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	16.0	46,590	3,963	8.51%	112
Standard_D4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	16.0	50,797	306	0.60%	28

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D8s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	32.0	89,102	849	0.95%	56
Standard_D8s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	32.0	89,422	6,441	7.20%	154
Standard_D8s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	32.0	85,673	2,704	3.16%	112
Standard_D8s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	32.0	101,753	1,013	1.00%	14
Standard_D16s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	1	64.0	179,390	1,403	0.78%	42
Standard_D16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	64.0	173,313	14,382	8.30%	98
Standard_D16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	64.0	171,750	1,261	0.73%	70
Standard_D16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	64.0	204,568	2,434	1.19%	14

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D32s_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	32	2	128.0	358,426	6,880	1.92%	56
Standard_D32s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	1	128.0	364,032	20,351	5.59%	84
Standard_D32s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	128.0	346,172	2,859	0.83%	84

### Dv3 - General Compute

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D2_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	8.0	23,795	1,893	7.96%	70
Standard_D2_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	8.0	24,582	2,036	8.28%	154
Standard_D2_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	8.0	24,376	1,915	7.86%	84
Standard_D4_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	16.0	45,883	3,929	8.56%	70

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D4_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	16.0	46,836	5,296	11.31%	140
Standard_D4_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	16.0	46,281	4,133	8.93%	112
Standard_D8_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	32.0	88,815	1,091	1.23%	126
Standard_D8_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	32.0	89,625	6,366	7.10%	112
Standard_D8_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	32.0	87,549	3,215	3.67%	98
Standard_D32_v3	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	32	2	128.0	353,069	3,792	1.07%	70
Standard_D32_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	1	128.0	358,984	19,517	5.44%	126
Standard_D32_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	128.0	356,479	16,176	4.54%	126

#### DSv2 - General Purpose + Premium Storage

(05/24/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	3.5	17,338	2,482	14.32%	98
Standard_DS1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	3.5	18,579	2,267	12.20%	112
Standard_DS1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	3.5	18,617	2,963	15.92%	168
Standard_DS1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	3.5	20,095	2,368	11.79%	154
Standard_DS2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	7.0	33,367	1,923	5.76%	98
Standard_DS2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	7.0	34,329	4,015	11.70%	182
Standard_DS2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	7.0	34,084	3,226	9.46%	154
Standard_DS2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	7.0	38,782	4,385	11.31%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS3_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	14.0	67,964	4,394	6.47%	112
Standard_DS3_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	14.0	67,396	5,434	8.06%	168
Standard_DS3_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	14.0	63,593	4,477	7.04%	112
Standard_DS3_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	14.0	78,599	8,145	10.36%	140
Standard_DS4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	28.0	129,452	4,586	3.54%	98
Standard_DS4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	28.0	128,545	6,193	4.82%	140
Standard_DS4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	28.0	125,892	6,614	5.25%	140
Standard_DS4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	28.0	150,995	10,036	6.65%	98

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS5_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	56.0	250,112	1,907	0.76%	84
Standard_DS5_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	56.0	263,493	11,774	4.47%	182
Standard_DS5_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	56.0	248,170	10,260	4.13%	126
Standard_DS5_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	56.0	304,566	25,421	8.35%	98

## Dv2 - General Compute

(05/24/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	3.5	17,093	1,249	7.30%	224
Standard_D1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	3.5	17,229	1,954	11.34%	182
Standard_D1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	3.5	16,474	1,558	9.46%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	3.5	21,631	306	1.41%	14
Standard_D2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	7.0	33,613	1,961	5.83%	196
Standard_D2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	7.0	34,301	2,859	8.34%	168
Standard_D2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	7.0	34,041	2,386	7.01%	140
Standard_D2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	7.0	37,309	911	2.44%	14
Standard_D3_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	14.0	65,186	3,256	4.99%	196
Standard_D3_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	14.0	66,161	3,850	5.82%	168
Standard_D3_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	14.0	65,731	4,742	7.21%	126

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D3_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	14.0	78,383	5,224	6.67%	28
Standard_D4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	28.0	129,637	5,291	4.08%	238
Standard_D4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	28.0	126,313	2,045	1.62%	182
Standard_D4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	28.0	120,903	1,877	1.55%	84
Standard_D4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	28.0	145,529	1,683	1.16%	14
Standard_D5_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	56.0	253,120	9,301	3.67%	238
Standard_D5_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	56.0	258,915	10,136	3.91%	126
Standard_D5_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	56.0	242,876	4,157	1.71%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D5_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	56.0	284,831	20,223	7.10%	28

### Av2 - General Compute

(04/12/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_A1_v2	Intel(R) Xeon(R) CPU E5-2660 v0 @ 2.20GHz	1	1	2.0	6,854	551	8.04%	28
Standard_A1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	2.0	6,798	724	10.65%	140
Standard_A1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	2.0	6,476	1,151	17.77%	84
Standard_A1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	2.0	6,195	397	6.41%	56
Standard_A1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	2.0	5,924	674	11.38%	28
Standard_A2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	4.0	13,666	1,079	7.89%	168

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_A2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	4.0	14,276	756	5.29%	84
Standard_A2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	4.0	14,223	1,010	7.10%	70
Standard_A2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	4.0	14,805	166	1.12%	14
Standard_A2m_v2	Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz	2	1	16.0	15,159	72	0.47%	14
Standard_A2m_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	16.0	14,315	1,255	8.76%	126
Standard_A2m_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	16.0	14,149	1,121	7.92%	98
Standard_A2m_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	16.0	13,791	1,201	8.71%	98
Standard_A2m_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	11,336	898	7.93%	14

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_A4_v2	Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz	4	1	8.0	28,551	1,839	6.44%	42
Standard_A4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	8.0	27,710	2,001	7.22%	98
Standard_A4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	8.0	28,371	2,769	9.76%	98
Standard_A4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	8.0	26,780	2,141	7.99%	84
Standard_A4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	8.0	26,476	1,608	6.07%	14
Standard_A4m_v2	Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz	4	1	32.0	28,710	1,032	3.59%	14
Standard_A4m_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	32.0	28,128	2,283	8.12%	112
Standard_A4m_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	32.0	28,220	2,997	10.62%	126

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_A4m_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	32.0	27,720	1,839	6.63%	98
Standard_A8_v2	Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz	8	1	16.0	52,295	1,356	2.59%	28
Standard_A8_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	16.0	56,004	4,045	7.22%	140
Standard_A8_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	16.0	53,367	3,252	6.09%	84
Standard_A8_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	16.0	51,525	3,170	6.15%	84
Standard_A8_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	16.0	49,802	1,009	2.03%	14
Standard_A8m_v2	Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz	8	2	64.0	51,496	1,643	3.19%	14
Standard_A8m_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	64.0	54,477	3,024	5.55%	126

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_A8m_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	64.0	55,195	4,426	8.02%	126
Standard_A8m_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	64.0	52,768	3,528	6.69%	70
Standard_A8m_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	49,188	1,869	3.80%	14

## High performance compute

### HBS - memory bandwidth (AMD EPYC)

(04/29/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HB60rs	AMD EPYC 7551 32-Core Processor	60	15	228.0	1,023,210	20,154	1.97%	42

### HCS - dense computation (Intel Xeon Platinum 8168)

(04/28/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_HC44rs	Intel(R) Xeon(R) Platinum 8168 CPU @ 2.70GHz	44	2	352.0	963,560	17,319	1.80%	84

## Memory optimized

### Easv4

(03/26/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2as_v4	AMD EPYC 7452 32-Core Processor	2	1	16.0	38,070	1,150	3.02%	210
Standard_E4as_v4	AMD EPYC 7452 32-Core Processor	4	1	32.0	75,733	1,444	1.91%	196
Standard_E4-2as_v4	AMD EPYC 7452 32-Core Processor	2	1	32.0	38,105	943	2.47%	168
Standard_E8as_v4	AMD EPYC 7452 32-Core Processor	8	1	64.0	149,522	2,333	1.56%	210
Standard_E8-2as_v4	AMD EPYC 7452 32-Core Processor	2	1	64.0	38,103	1,078	2.83%	168
Standard_E8-4as_v4	AMD EPYC 7452 32-Core Processor	4	1	64.0	76,060	1,132	1.49%	168
Standard_E16as_v4	AMD EPYC 7452 32-Core Processor	16	2	128.0	288,136	4,720	1.64%	210
Standard_E16-4as_v4	AMD EPYC 7452 32-Core Processor	4	2	128.0	73,038	2,310	3.16%	196
Standard_E16-8as_v4	AMD EPYC 7452 32-Core Processor	8	2	128.0	144,266	2,782	1.93%	168

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E20as_v4	AMD EPYC 7452 32-Core Processor	20	3	160.0	346,277	7,387	2.13%	14
Standard_E20as_v4	AMD EPYC 7452 32-Core Processor	20	5	160.0	351,213	7,002	1.99%	196
Standard_E32as_v4	AMD EPYC 7452 32-Core Processor	32	4	256.0	561,950	7,679	1.37%	42
Standard_E32-8as_v4	AMD EPYC 7452 32-Core Processor	8	4	256.0	143,569	3,393	2.36%	182
Standard_E32-16as_v4	AMD EPYC 7452 32-Core Processor	16	4	256.0	283,614	5,018	1.77%	182
Standard_E48as_v4	AMD EPYC 7452 32-Core Processor	48	6	384.0	832,627	19,565	2.35%	210
Standard_E64as_v4	AMD EPYC 7452 32-Core Processor	64	8	512.0	1,097,588	26,100	2.38%	280
Standard_E64-16as_v4	AMD EPYC 7452 32-Core Processor	16	8	512.0	284,934	5,065	1.78%	154
Standard_E64-32as_v4	AMD EPYC 7452 32-Core Processor	32	8	512.0	561,951	9,691	1.72%	140

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E96as_v4	AMD EPYC 7452 32-Core Processor	96	12	672.0	N/A	-	-	-
Standard_E96-24as_v4	AMD EPYC 7452 32-Core Processor	24	11	672.0	423,442	8,504	2.01%	182
Standard_E96-48as_v4	AMD EPYC 7452 32-Core Processor	48	11	672.0	839,993	14,218	1.69%	70

#### Eav4

(03/27/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2a_v4	AMD EPYC 7452 32-Core Processor	2	1	16.0	38,008	995	2.62%	210
Standard_E4a_v4	AMD EPYC 7452 32-Core Processor	4	1	32.0	75,410	1,431	1.90%	196
Standard_E8a_v4	AMD EPYC 7452 32-Core Processor	8	1	64.0	148,810	2,630	1.77%	210
Standard_E16a_v4	AMD EPYC 7452 32-Core Processor	16	2	128.0	286,811	4,877	1.70%	182
Standard_E20a_v4	AMD EPYC 7452 32-Core Processor	20	3	160.0	351,049	6,268	1.79%	210

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32a_v4	AMD EPYC 7452 32-Core Processor	32	4	256.0	565,363	10,941	1.94%	126
Standard_E48a_v4	AMD EPYC 7452 32-Core Processor	48	6	384.0	837,493	15,803	1.89%	126
Standard_E64a_v4	AMD EPYC 7452 32-Core Processor	64	8	512.0	1,097,111	30,290	2.76%	336
Standard_E96a_v4	AMD EPYC 7452 32-Core Processor	96	12	672.0	N/A	-	-	-

## EDSv4

(03/27/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	34,923	1,107	3.17%	336
Standard_E4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	66,921	1,294	1.93%	322
Standard_E4-2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	32.0	34,909	811	2.32%	294

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	132,164	2,102	1.59%	154
Standard_E8-2ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	64.0	35,031	965	2.76%	252
Standard_E8-4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	64.0	67,144	1,200	1.79%	182
Standard_E16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	265,181	2,634	0.99%	336
Standard_E16-4ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	128.0	67,155	1,596	2.38%	336
Standard_E16-8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	128.0	132,939	1,471	1.11%	336
Standard_E20ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	331,456	2,766	0.83%	336
Standard_E32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	531,560	5,700	1.07%	196

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	256.0	512,931	5,110	1.00%	14
Standard_E32-8ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	256.0	132,929	1,671	1.26%	182
Standard_E32-16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	256.0	265,471	2,268	0.85%	154
Standard_E48ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	768,428	6,891	0.90%	224
Standard_E64ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,005,554	78,398	7.80%	140
Standard_E64-16ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	504.0	260,677	3,340	1.28%	154
Standard_E64-32ds_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	504.0	514,504	4,082	0.79%	98

### Edsv4 Isolated Extended

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E80ids_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	80	2	504.0	N/A	-	-	-

## EDv4

(03/26/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	34,916	1,063	3.04%	322
Standard_E4d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	66,889	1,283	1.92%	336
Standard_E8d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	132,382	2,020	1.53%	322
Standard_E16d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	265,094	2,803	1.06%	336
Standard_E20d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	331,516	2,568	0.77%	336
Standard_E32d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	530,364	9,914	1.87%	336

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E48d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	761,410	21,640	2.84%	336
Standard_E64d_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	1,030,708	9,500	0.92%	322

#### EIASv4

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E96ias_v4	AMD EPYC 7452 32-Core Processor	96	12	672.0	N/A	-	-	-

#### Esv4

(03/25/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	31,390	2,786	8.88%	336
Standard_E4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	59,677	3,904	6.54%	336
Standard_E4-2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	32.0	31,443	2,480	7.89%	364

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	117,898	7,464	6.33%	406
Standard_E8-2s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	64.0	30,989	2,864	9.24%	406
Standard_E8-4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	64.0	59,589	4,762	7.99%	406
Standard_E16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	236,972	13,376	5.64%	406
Standard_E16-4s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	128.0	60,316	4,792	7.94%	406
Standard_E16-8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	128.0	117,057	6,569	5.61%	392
Standard_E20s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	294,231	15,477	5.26%	406
Standard_E32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	481,943	22,707	4.71%	266

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32-8s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	256.0	116,774	6,791	5.82%	224
Standard_E32-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	256.0	235,620	11,909	5.05%	266
Standard_E32-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	256.0	222,478	3,411	1.53%	14
Standard_E48s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	693,841	23,265	3.35%	182
Standard_E64s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	922,196	7,708	0.84%	182
Standard_E64-16s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	2	504.0	224,499	3,955	1.76%	168
Standard_E64-32s_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	2	504.0	441,521	30,939	7.01%	168

#### Esv4 Isolated Extended

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
---------	-----	-------	------------	--------------	-----------	--------	---------	-------

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E80is_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	80	2	504.0	N/A	-	-	-

#### Ev4

(03/25/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	30,825	2,765	8.97%	406
Standard_E4_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	60,495	4,419	7.30%	406
Standard_E8_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	64.0	119,562	8,628	7.22%	406
Standard_E16_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	237,126	13,328	5.62%	392
Standard_E20_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	299,681	17,288	5.77%	406
Standard_E32_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	486,051	28,085	5.78%	378

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E48_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	48	2	384.0	686,812	20,561	2.99%	378
Standard_E64_v4	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	64	2	504.0	919,491	15,261	1.66%	378

### Esv3 - Memory Optimized + Premium Storage

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	16.0	23,704	2,155	9.09%	168
Standard_E2s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	16.0	21,917	1,521	6.94%	112
Standard_E2s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	28,549	3,105	10.88%	42
Standard_E4s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	32.0	46,370	4,256	9.18%	140
Standard_E4s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	32.0	47,178	3,791	8.04%	98

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E4s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	53,636	4,231	7.89%	84
Standard_E16s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	128.0	175,905	7,275	4.14%	196
Standard_E16s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	128.0	176,579	9,650	5.47%	112
Standard_E16s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	206,776	19,901	9.62%	28
Standard_E20s_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	160.0	219,370	7,086	3.23%	224
Standard_E20s_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	160.0	224,353	11,954	5.33%	98
Standard_E20s_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	280,572	13,326	4.75%	28

### Ev3 - Memory Optimized

(04/05/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E2_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	16.0	23,304	2,074	8.90%	182
Standard_E2_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	16.0	24,513	2,428	9.90%	112
Standard_E2_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	16.0	26,171	153	0.58%	14
Standard_E4_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	32.0	46,224	3,713	8.03%	238
Standard_E4_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	32.0	49,200	3,457	7.03%	42
Standard_E4_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	32.0	53,476	4,219	7.89%	42
Standard_E8_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	64.0	90,915	7,711	8.48%	224
Standard_E8_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	64.0	89,968	5,738	6.38%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E16_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	128.0	174,677	7,198	4.12%	210
Standard_E16_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	128.0	180,002	14,028	7.79%	98
Standard_E16_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	128.0	217,439	13,826	6.36%	28
Standard_E20_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	160.0	221,787	10,447	4.71%	238
Standard_E20_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	160.0	234,854	10,704	4.56%	70
Standard_E20_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	160.0	293,226	3,480	1.19%	14
Standard_E32_v3	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	32	2	256.0	349,134	13,895	3.98%	210
Standard_E32_v3	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	32	1	256.0	352,509	14,689	4.17%	84

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_E32_v3	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	32	1	256.0	413,946	2,239	0.54%	14

### DSv2 - Memory Optimized + Premium Storage

(05/24/2021 PBID:9198755 OS: MicrosoftWindowsServer-WindowsServer-2019-Datacenter-latest)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS11_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	14.0	33,150	2,097	6.33%	112
Standard_DS11_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	14.0	34,294	3,107	9.06%	182
Standard_DS11_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	14.0	33,447	2,690	8.04%	98
Standard_DS11_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	14.0	37,222	3,740	10.05%	126
Standard_DS11-1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	14.0	16,994	1,127	6.63%	98
Standard_DS11-1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	14.0	17,538	2,029	11.57%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS11-1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	14.0	17,255	1,647	9.54%	140
Standard_DS11-1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	14.0	19,463	3,017	15.50%	140
Standard_DS12_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	28.0	65,209	2,830	4.34%	98
Standard_DS12_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	28.0	70,755	5,315	7.51%	140
Standard_DS12_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	28.0	62,898	3,666	5.83%	126
Standard_DS12_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	28.0	76,876	7,628	9.92%	140
Standard_DS12-1_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	1	1	28.0	17,253	1,094	6.34%	56
Standard_DS12-1_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	1	1	28.0	17,748	2,342	13.20%	182

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS12-1_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	1	1	28.0	16,866	1,506	8.93%	154
Standard_DS12-1_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	1	1	28.0	18,925	2,461	13.00%	140
Standard_DS12-2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	28.0	35,288	2,477	7.02%	98
Standard_DS12-2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	28.0	35,716	2,923	8.19%	140
Standard_DS12-2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	28.0	33,884	3,285	9.70%	112
Standard_DS12-2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	28.0	39,473	3,712	9.40%	140
Standard_DS13_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	56.0	136,322	6,996	5.13%	70
Standard_DS13_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	56.0	127,110	4,330	3.41%	154

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS13_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	56.0	126,037	6,952	5.52%	126
Standard_DS13_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	56.0	158,319	11,403	7.20%	140
Standard_DS13-2_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	56.0	34,673	1,705	4.92%	56
Standard_DS13-2_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	56.0	34,869	4,134	11.86%	182
Standard_DS13-2_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	56.0	33,986	2,411	7.09%	112
Standard_DS13-2_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	56.0	39,104	3,607	9.22%	140
Standard_DS13-4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	56.0	67,217	3,774	5.61%	84
Standard_DS13-4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	56.0	69,377	6,103	8.80%	210

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS13-4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	56.0	66,193	6,252	9.45%	70
Standard_DS13-4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	56.0	76,528	6,662	8.71%	126
Standard_DS14_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	112.0	255,718	11,055	4.32%	98
Standard_DS14_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	112.0	263,225	11,664	4.43%	182
Standard_DS14_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	112.0	257,737	11,726	4.55%	84
Standard_DS14_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	112.0	296,962	22,427	7.55%	126
Standard_DS14-4_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	2	112.0	67,170	4,554	6.78%	70
Standard_DS14-4_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	112.0	66,748	4,294	6.43%	140

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS14-4_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	112.0	67,116	7,254	10.81%	140
Standard_DS14-4_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	112.0	75,100	4,719	6.28%	154
Standard_DS14-8_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	2	112.0	125,558	1,961	1.56%	84
Standard_DS14-8_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	112.0	128,904	4,055	3.15%	84
Standard_DS14-8_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	112.0	127,309	8,301	6.52%	182
Standard_DS14-8_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	112.0	148,717	13,028	8.76%	140
Standard_DS15_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	140.0	314,358	3,189	1.01%	56
Standard_DS15_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	140.0	324,517	14,805	4.56%	168

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_DS15_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	140.0	312,229	16,417	5.26%	126
Standard_DS15_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	140.0	369,680	36,859	9.97%	126
Standard_DS15i_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	140.0	330,307	18,294	5.54%	28
Standard_DS15i_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	140.0	330,397	17,469	5.29%	294
Standard_DS15i_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	140.0	371,086	33,246	8.96%	140

## Dv2 - Memory Optimized

(05/24/2021 PBID:9198755 OS: MicrosoftWindowsServer-WindowsServer-2019-Datacenter-latest)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D11_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	2	1	14.0	33,085	1,835	5.55%	196
Standard_D11_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	2	1	14.0	33,998	3,523	10.36%	168

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D11_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	2	1	14.0	32,964	2,690	8.16%	140
Standard_D11_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	2	1	14.0	39,330	3,092	7.86%	28
Standard_D12_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	4	1	28.0	65,877	3,537	5.37%	196
Standard_D12_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	4	1	28.0	66,128	4,438	6.71%	210
Standard_D12_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	4	1	28.0	63,397	4,021	6.34%	98
Standard_D12_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	4	1	28.0	80,559	3,760	4.67%	14
Standard_D13_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	8	1	56.0	133,275	7,394	5.55%	182
Standard_D13_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	8	1	56.0	127,557	3,854	3.02%	210

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GiB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D13_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	8	1	56.0	120,793	1,444	1.20%	70
Standard_D13_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	8	1	56.0	151,473	7,667	5.06%	42
Standard_D14_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	16	2	112.0	253,422	9,024	3.56%	224
Standard_D14_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	16	1	112.0	261,437	12,701	4.86%	140
Standard_D14_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	16	1	112.0	246,994	6,166	2.50%	98
Standard_D14_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	16	1	112.0	315,200	15,164	4.81%	28
Standard_D15_v2	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	20	2	140.0	315,711	8,560	2.71%	252
Standard_D15_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	140.0	321,475	12,246	3.81%	168

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_D15_v2	Intel(R) Xeon(R) Platinum 8171M CPU @ 2.60GHz	20	1	140.0	314,692	11,528	3.66%	56
Standard_D15i_v2	Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30GHz	20	1	140.0	323,411	10,041	3.10%	462
Standard_D15i_v2	Intel(R) Xeon(R) Platinum 8272CL CPU @ 2.60GHz	20	1	140.0	370,436	5,018	1.35%	28

## Storage optimized

### Lsv2 - Storage Optimized

(04/29/2021 PBID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_L8s_v2	AMD EPYC 7551 32-Core Processor	8	1	64.0	101,433	844	0.83%	448
Standard_L16s_v2	AMD EPYC 7551 32-Core Processor	16	2	128.0	200,664	2,866	1.43%	448
Standard_L32s_v2	AMD EPYC 7551 32-Core Processor	32	4	256.0	396,781	7,237	1.82%	462
Standard_L48s_v2	AMD EPYC 7551 32-Core Processor	48	6	384.0	584,059	13,101	2.24%	112

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_L64s_v2	AMD EPYC 7551 32-Core Processor	64	8	512.0	768,442	19,518	2.54%	70

## Ls - Storage Optimized + Premium Storage

(05/25/2021 PBIID:9198755)

VM SIZE	CPU	VCPUS	NUMA NODES	MEMORY (GIB)	AVG SCORE	STDDEV	STDDEV%	#RUNS
Standard_L4s	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00GHz	4	1	32.0	69,902	6,012	8.60%	294
Standard_L8s	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00GHz	8	1	64.0	133,859	7,026	5.25%	280
Standard_L16s	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00GHz	16	1	128.0	259,512	7,269	2.80%	252
Standard_L32s	Intel(R) Xeon(R) CPU E5-2698B v3 @ 2.00GHz	32	2	256.0	506,359	7,472	1.48%	252

## About CoreMark

[CoreMark](#) is a benchmark that tests the functionality of a microcontroller (MCU) or central processing unit (CPU). CoreMark isn't system dependent, so it functions the same regardless of the platform (for example, big or little endian, high-end or low-end processor).

Windows numbers were computed by running CoreMark on Windows Server 2019. CoreMark was configured with the number of threads set to the number of virtual CPUs, and concurrency set to `PThreads`. The target number of iterations was adjusted based on expected performance to provide a runtime of at least 20 seconds (typically much longer). The final score represents the number of iterations completed divided by the number of seconds it took to run the test. Each test was run at least seven times on each VM. Test run dates shown above. Tests run on multiple VMs across Azure public regions the VM was supported in on the date run.

Windows numbers were computed by running CoreMark on Windows Server 2019. CoreMark was configured with the number of threads set to the number of virtual CPUs, and concurrency set to `PThreads`. The target

number of iterations was adjusted based on expected performance to provide a runtime of at least 20 seconds (typically much longer). The final score represents the number of iterations completed divided by the number of seconds it took to run the test. Each test was run at least seven times on each VM. Test run dates shown above. Tests run on multiple VMs across Azure public regions the VM was supported in on the date run. (Coremark doesn't properly support more than 64 vCPUs on Windows, therefore SKUs with > 64 vCPUs have been marked as N/A.)

## Running Coremark on Azure VMs

### Download:

CoreMark is an open source tool that can be downloaded from [GitHub](#).

### Building and Running:

To build and run the benchmark, type:

```
> make
```

Full results are available in the files `run1.log` and `run2.log`. `run1.log` contains CoreMark results with performance parameters. `run2.log` contains benchmark results with validation parameters.

### Run Time:

By default, the benchmark will run between 10-100 seconds. To override, use `ITERATIONS=N`

```
% make ITERATIONS=10
```

above flag will run the benchmark for 10 iterations. **Results are only valid for reporting if the benchmark ran for at least 10 seconds!**

### Parallel Execution:

Use `XCFLAGS=-DMULTITHREAD=N` where N is number of threads to run in parallel. Several implementations are available to execute in multiple contexts.

```
% make XCFLAGS="-DMULTITHREAD=4 -DUSE_PTHREAD"
```

The above will compile the benchmark for execution on 4 cores.

### Recommendations for best results

- The benchmark needs to run for at least 10 seconds, probably longer on larger systems.
- All source files must be compiled with same flags.
- Don't change source files other than `core_portme*` (use `make check` to validate)
- Multiple runs are suggested for best results.

## GPU Series

Performance of GPU based VM series is best understood by using GPU appropriate benchmarks and running at the scale required for your workloads. Azure ranks among the best there:

- Top 10 Supercomputer: [November 2021 | TOP500](#) (Azure powered #10: Voyager-EUS2)
- Machine Learning: MLCommons Training: [v1.1 Results | MLCommons](#) (2 highest at scale and largest in the cloud)

## Next steps

- For storage capacities, disk details, and other considerations for choosing among VM sizes, see [Sizes for virtual machines](#).

# Check vCPU quotas using the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

The vCPU quotas for virtual machines and virtual machine scale sets are arranged in two tiers for each subscription, in each region. The first tier is the Total Regional vCPUs, and the second tier is the various VM size family cores such as the D-series vCPUs. Any time a new VM is deployed the vCPUs for the VM must not exceed the vCPU quota for the VM size family or the total regional vCPU quota. If either of those quotas are exceeded, the VM deployment will not be allowed. There is also a quota for the overall number of virtual machines in the region. The details on each of these quotas can be seen in the **Usage + quotas** section of the **Subscription** page in the [Azure portal](#), or you can query for the values using the Azure CLI.

## NOTE

Quota is calculated based on the total number of cores in use both allocated and deallocated. If you need additional cores, [request a quota increase](#) or delete VMs that are no longer needed.

## Check usage

You can check your quota usage using [az vm list-usage](#).

```
az vm list-usage --location "East US" -o table
```

The output should look something like this:

Name	CurrentValue	Limit
Availability Sets	0	2000
Total Regional vCPUs	29	100
Virtual Machines	7	10000
Virtual Machine Scale Sets	0	2000
Standard DSv3 Family vCPUs	8	100
Standard DSv2 Family vCPUs	3	100
Standard Dv3 Family vCPUs	2	100
Standard D Family vCPUs	8	100
Standard Dv2 Family vCPUs	8	100
Basic A Family vCPUs	0	100
Standard A0-A7 Family vCPUs	0	100
Standard A8-A11 Family vCPUs	0	100
Standard DS Family vCPUs	0	100
Standard G Family vCPUs	0	100
Standard GS Family vCPUs	0	100
Standard F Family vCPUs	0	100
Standard FS Family vCPUs	0	100
Standard Storage Managed Disks	5	10000
Premium Storage Managed Disks	5	10000

## Reserved VM Instances

Reserved VM Instances, which are scoped to a single subscription without VM size flexibility, will add a new aspect to the vCPU quotas. These values describe the number of instances of the stated size that must be

deployable in the subscription. They work as a placeholder in the quota system to ensure that quota is reserved to ensure Azure reservations are deployable in the subscription. For example, if a specific subscription has 10 Standard\_D1 reservations the usages limit for Standard\_D1 reservations will be 10. This will cause Azure to ensure that there are always at least 10 vCPUs available in the Total Regional vCPUs quota to be used for Standard\_D1 instances and there are at least 10 vCPUs available in the Standard D Family vCPU quota to be used for Standard\_D1 instances.

If a quota increase is required to either purchase a Single Subscription RI, you can [request a quota increase on your subscription](#).

## Next steps

For more information about billing and quotas, see [Azure subscription and service limits, quotas, and constraints](#).

# Check vCPU quotas using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

The vCPU quotas for virtual machines and virtual machine scale sets are arranged in two tiers for each subscription, in each region. The first tier is the Total Regional vCPUs, and the second tier is the various VM size family cores such as the D-series vCPUs. Any time a new VM is deployed the vCPUs for the VM must not exceed the vCPU quota for the VM size family or the total regional vCPU quota. If either of those quotas are exceeded, the VM deployment will not be allowed. There is also a quota for the overall number of virtual machines in the region. The details on each of these quotas can be seen in the **Usage + quotas** section of the **Subscription** page in the [Azure portal](#), or you can query for the values using PowerShell.

## NOTE

Quota is calculated based on the total number of cores in use both allocated and deallocated. If you need additional cores, [request a quota increase](#) or delete VMs that are no longer needed.

## Check usage

You can use the [Get-AzVMUsage](#) cmdlet to check on your quota usage.

```
Get-AzVMUsage -Location "East US"
```

The output will look similar to this:

Name	Current Value	Limit	Unit
---	-----	-----	-----
Availability Sets	0	2000	Count
Total Regional vCPUs	4	260	Count
Virtual Machines	4	10000	Count
Virtual Machine Scale Sets	1	2000	Count
Standard B Family vCPUs	1	10	Count
Standard DSv2 Family vCPUs	1	100	Count
Standard Dv2 Family vCPUs	2	100	Count
Basic A Family vCPUs	0	100	Count
Standard A0-A7 Family vCPUs	0	250	Count
Standard A8-A11 Family vCPUs	0	100	Count
Standard D Family vCPUs	0	100	Count
Standard G Family vCPUs	0	100	Count
Standard DS Family vCPUs	0	100	Count
Standard GS Family vCPUs	0	100	Count
Standard F Family vCPUs	0	100	Count
Standard FS Family vCPUs	0	100	Count
Standard NV Family vCPUs	0	24	Count
Standard NC Family vCPUs	0	48	Count
Standard H Family vCPUs	0	8	Count
Standard Av2 Family vCPUs	0	100	Count
Standard LS Family vCPUs	0	100	Count
Standard Dv2 Promo Family vCPUs	0	100	Count
Standard DSv2 Promo Family vCPUs	0	100	Count
Standard MS Family vCPUs	0	0	Count
Standard Dv3 Family vCPUs	0	100	Count
Standard DSv3 Family vCPUs	0	100	Count
Standard Ev3 Family vCPUs	0	100	Count
Standard ESv3 Family vCPUs	0	100	Count
Standard FSv2 Family vCPUs	0	100	Count
Standard ND Family vCPUs	0	0	Count
Standard NCv2 Family vCPUs	0	0	Count
Standard NCv3 Family vCPUs	0	0	Count
Standard LSv2 Family vCPUs	0	0	Count
Standard Storage Managed Disks	2	10000	Count
Premium Storage Managed Disks	1	10000	Count

## Reserved VM Instances

Reserved VM Instances, which are scoped to a single subscription without VM size flexibility, will add a new aspect to the vCPU quotas. These values describe the number of instances of the stated size that must be deployable in the subscription. They work as a placeholder in the quota system to ensure that quota is reserved to ensure reserved VM instances are deployable in the subscription. For example, if a specific subscription has 10 Standard\_D1 reserved VM instances the usages limit for Standard\_D1 reserved VM instances will be 10. This will cause Azure to ensure that there are always at least 10 vCPUs available in the Total Regional vCPUs quota to be used for Standard\_D1 instances and there are at least 10 vCPUs available in the Standard D Family vCPU quota to be used for Standard\_D1 instances.

If a quota increase is required to purchase a Single Subscription RI, you can [request a quota increase](#) on your subscription.

## Next steps

For more information about billing and quotas, see [Azure subscription and service limits, quotas, and constraints](#).

# Change the size of a virtual machine

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article shows you how to move a VM to a different [VM size](#).

After you create a virtual machine (VM), you can scale the VM up or down by changing the VM size. In some cases, you must deallocate the VM first. This can happen if the new size is not available on the hardware cluster that is currently hosting the VM.

If your VM uses Premium Storage, make sure that you choose an s version of the size to get Premium Storage support. For example, choose Standard\_E4s\_v3 instead of Standard\_E4\_v3.

## Change the VM size

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Open the [Azure portal](#).
2. Open the page for the virtual machine.
3. In the left menu, select **Size**.
4. Pick a new size from the list of available sizes and then select **Resize**.

If the virtual machine is currently running, changing its size will cause it to be restarted.

If your VM is still running and you don't see the size you want in the list, stopping the virtual machine may reveal more sizes.

### WARNING

Deallocating the VM also releases any dynamic IP addresses assigned to the VM. The OS and data disks are not affected.

If you are resizing a production VM, consider using [Azure Capacity Reservations](#) to reserve Compute capacity in the region.

## Next steps

For additional scalability, run multiple VM instances and scale out. For more information, see [Automatically scale machines in a Virtual Machine Scale Set](#).

# States and billing status of Azure Virtual Machines

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure Virtual Machines (VM) instances go through different states. There are *provisioning* and *power* states. This article describes these states and highlights when customers are billed for instance usage.

## Get states using Instance View

The instance view API provides VM running-state information. For more information, see [Virtual Machines - Instance View](#).

Azure Resources Explorer provides a simple UI for viewing the VM running state: [Resource Explorer](#).

The VM provisioning state is available, in slightly different forms, from within the VM properties `provisioningState` and the `InstanceView`. In the VM `InstanceView`, there's an element within the `status` array in the form of `ProvisioningState/<state>[/<errorCode>]`.

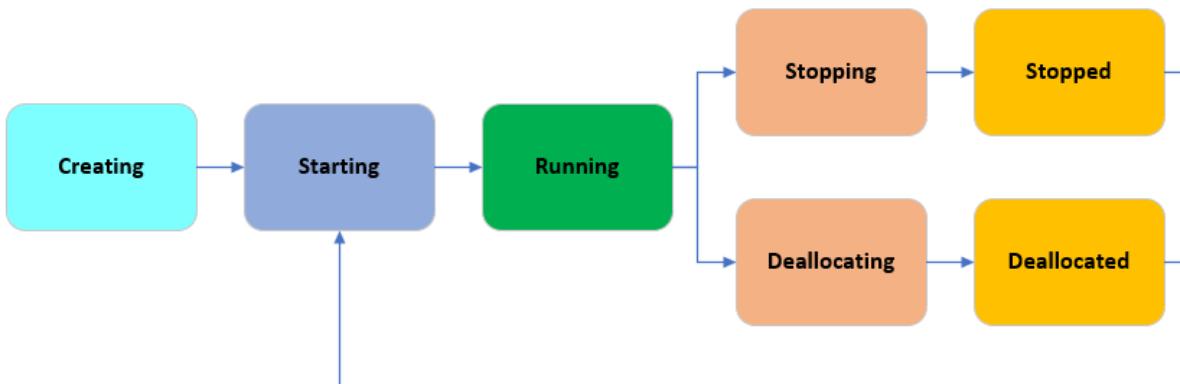
To retrieve the power state of all the VMs in your subscription, use the [Virtual Machines - List All API](#) with parameter `statusOnly` set to `true`.

### NOTE

[Virtual Machines - List All API](#) with parameter `statusOnly` set to `true` retrieves the power states of all VMs in a subscription. However, in some rare situations, the power state may not be available due to intermittent issues in the retrieval process. In such situations, we recommend retrying using the same API or using [Azure Resource Health](#) to check the power state of your VMs.

## Power states and billing

The power state represents the last known state of the VM.



The following table provides a description of each instance state and indicates whether that state is billed for instance usage.

POWER STATE	DESCRIPTION	BILLING
Creating	Virtual machine is allocating resources.	Not Billed*

POWER STATE	DESCRIPTION	BILLING
Starting	Virtual machine is powering up.	Billed
Running	Virtual machine is fully up. This state is the standard working state.	Billed
Stopping	This state is transitional between running and stopped.	Billed
Stopped	The virtual machine is allocated on a host but not running. Also called <i>PoweredOff</i> state or <i>Stopped (Allocated)</i> . This state can be result of invoking the <code>Poweroff</code> API operation or invoking shutdown from within the guest OS. The <i>Stopped</i> state may also be observed briefly during VM creation or while starting a VM from <i>Deallocated</i> state.	Billed
Deallocating	This state is transitional between <i>Running</i> and <i>Deallocated</i> .	Not billed*
Deallocated	The virtual machine has released the lease on the underlying hardware and is powered off. This state is also referred to as <i>Stopped (Deallocated)</i> .	Not billed*

\* Some Azure resources, such as [Disks](#) and [Networking](#) continue to incur charges.

Example of PowerState in JSON:

```
{
  "code": "PowerState/running",
  "level": "Info",
  "displayStatus": "VM running"
}
```

## Provisioning states

The provisioning state is the status of a user-initiated, control-plane operation on the VM. These states are separate from the power state of a VM.

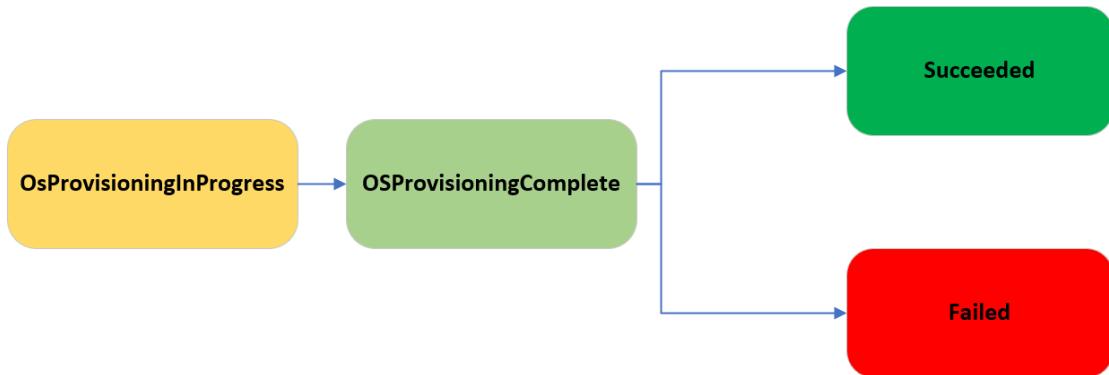
PROVISIONING STATE	DESCRIPTION
Creating	Virtual machine is being created.
Updating	Virtual machine is updating to the latest model. Some non-model changes to a virtual machine such as start and restart fall under the updating state.
Failed	Last operation on the virtual machine resource was unsuccessful.

PROVISIONING STATE	DESCRIPTION
Succeeded	Last operation on the virtual machine resource was successful.
Deleting	Virtual machine is being deleted.
Migrating	Seen when migrating from Azure Service Manager to Azure Resource Manager.

## OS Provisioning states

OS Provisioning states only apply to virtual machines created with a [generalized](#) OS image. [Specialized](#) images and disks attached as OS disk don't display these states. The OS provisioning state isn't shown separately. It's a substate of the Provisioning State in the VM InstanceView. For example,

```
ProvisioningState/creating/osProvisioningComplete .
```



OS PROVISIONING STATE	DESCRIPTION
OSProvisioningInProgress	The VM is running and the initialization (setup) of the Guest OS is in progress.
OSProvisioningComplete	This state is a short-lived state. The virtual machine quickly transitions from this state to <i>Success</i> . If extensions are still being installed, you continue to see this state until installation is complete.
Succeeded	The user-initiated actions have completed.
Failed	Represents a failed operation. For more information and possible solutions, see the error code.

## Troubleshooting VM states

To troubleshoot specific VM state issues, see [Troubleshoot Windows VM deployments](#) and [Troubleshoot Linux VM deployments](#).

For other troubleshooting help visit [Azure Virtual Machines troubleshooting documentation](#).

## Next steps

- Review the [Azure Cost Management and Billing documentation](#)
- Use the [Azure Pricing calculator](#) to plan your deployments.
- Learn more about monitoring your VM, see [Monitor virtual machines in Azure](#).



# Azure virtual machine sizes naming conventions

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This page outlines the naming conventions used for Azure VMs. VMs use these naming conventions to denote varying features and specifications.

## Naming convention explanation

[Family] + [Sub-family]\* + [# of vCPUs] + [Constrained vCPUs]\* + [Additive Features] + [Accelerator Type]\* + [Version]

VALUE	EXPLANATION
Family	Indicates the VM Family Series
*Sub-family	Used for specialized VM differentiations only
# of vCPUs	Denotes the number of vCPUs of the VM
*Constrained vCPUs	Used for certain VM sizes only. Denotes the number of vCPUs for the <a href="#">constrained vCPU capable size</a>
Additive Features	One or more lower case letters denote additive features, such as: a = AMD-based processor b = Block Storage performance c = confidential d = diskful (i.e., a local temp disk is present); this is for newer Azure VMs, see <a href="#">Ddv4 and Ddsv4-series</a> i = isolated size l = low memory; a lower amount of memory than the memory intensive size m = memory intensive; the most amount of memory in a particular size t = tiny memory; the smallest amount of memory in a particular size s = Premium Storage capable, including possible use of <a href="#">Ultra SSD</a> (Note: some newer sizes without the attribute of s can still support Premium Storage e.g. M128, M64, etc.) NP = node packing P = ARM Cpu
*Accelerator Type	Denotes the type of hardware accelerator in the specialized/GPU SKUs. Only the new specialized/GPU SKUs launched from Q3 2020 will have the hardware accelerator in the name.
Version	Denotes the version of the VM Family Series

## Example breakdown

[Family] + [Sub-family]\* + [# of vCPUs] + [Additive Features] + [Accelerator Type]\* + [Version]

**Example 1: M416ms\_v2**

VALUE	EXPLANATION
Family	M
# of vCPUs	416
Additive Features	m = memory intensive s = Premium Storage capable
Version	v2

**Example 2: NV16as\_v4**

VALUE	EXPLANATION
Family	N
Sub-family	V
# of vCPUs	16
Additive Features	a = AMD-based processor s = Premium Storage capable
Version	v4

**Example 3: NC4as\_T4\_v3**

VALUE	EXPLANATION
Family	N
Sub-family	C
# of vCPUs	4
Additive Features	a = AMD-based processor s = Premium Storage capable
Accelerator Type	T4
Version	v3

**Example 4: M8-2ms\_v2 (Constrained vCPU)**

VALUE	EXPLANATION
Family	M
# of vCPUs	8
# of constrained (actual) vCPUs	2

VALUE	EXPLANATION
Additive Features	m = memory intensive s = Premium Storage capable
Version	v2

## Next steps

Learn more about available [VM Sizes](#) in Azure.

# Store and share resources in an Azure Compute Gallery

9/21/2022 • 15 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

An Azure Compute Gallery helps you build structure and organization around your Azure resources, like images and [applications](#). An Azure Compute Gallery provides:

- Global replication.<sup>1</sup>
- Versioning and grouping of resources for easier management.
- Highly available resources with Zone Redundant Storage (ZRS) accounts in regions that support Availability Zones. ZRS offers better resilience against zonal failures.
- Premium storage support (Premium\_LRS).
- Sharing to the community, across subscriptions, and between Active Directory (AD) tenants.
- Scaling your deployments with resource replicas in each region.

With a gallery, you can share your resources to everyone, or limit sharing to different users, service principals, or AD groups within your organization. Resources can be replicated to multiple regions, for quicker scaling of your deployments.

<sup>1</sup> The Azure Compute Gallery service is not a global resource. For disaster recovery scenarios, it is a best practice is to have at least two galleries, in different regions.

## Images

For more information about storing images in an Azure Compute Gallery, see [Store and share images in an Azure Compute Gallery](#).

## VM apps

While you can create an image of a VM with apps pre-installed, you would need to update your image each time you have application changes. Separating your application installation from your VM images means there's no need to publish a new image for every line of code change.

For more information about storing applications in an Azure Compute Gallery, see [VM Applications](#).

## Regional Support

All public regions can be target regions, but certain regions require that customers go through a request process in order to gain access. To request that a subscription is added to the allowlist for a region such as Australia Central or Australia Central 2, submit [an access request](#)

## Limits

There are limits, per subscription, for deploying resources using Azure Compute Galleries:

- 100 galleries, per subscription, per region
- 1,000 image definitions, per subscription, per region
- 10,000 image versions, per subscription, per region

- 100 image version replicas, per subscription, per region however 50 replicas should be sufficient for most use cases
- Any disk attached to the image must be less than or equal to 1TB in size

For more information, see [Check resource usage against limits](#) for examples on how to check your current usage.

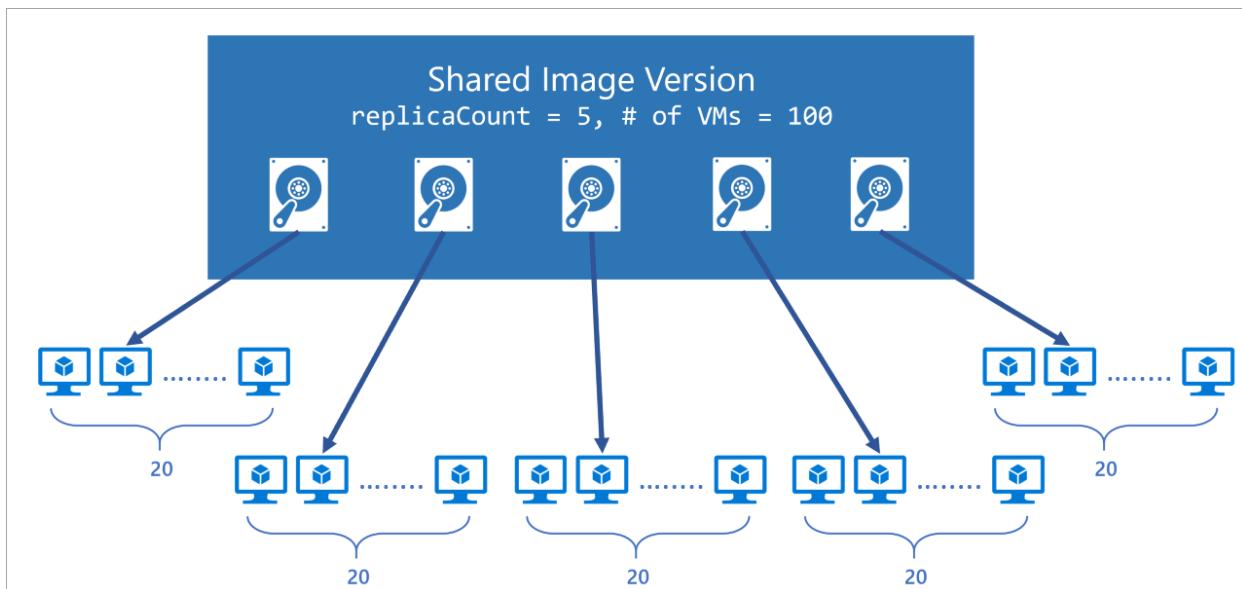
## Scaling

Azure Compute Gallery allows you to specify the number of replicas you want to keep. This helps in multi-VM deployment scenarios as the VM deployments can be spread to different replicas reducing the chance of instance creation processing being throttled due to overloading of a single replica.

With Azure Compute Gallery, you can deploy up to a 1,000 VM instances in a virtual machine scale set. You can set a different replica count in each target region, based on the scale needs for the region. Since each replica is a copy of your resource, this helps scale your deployments linearly with each extra replica. While we understand no two resources or regions are the same, here's our general guideline on how to use replicas in a region:

- For every 20 VMs that you create concurrently, we recommend you keep one replica. For example, if you are creating 120 VMs concurrently using the same image in a region, we suggest you keep at least 6 replicas of your image.
- For each scale set you create concurrently, we recommend you keep one replica.

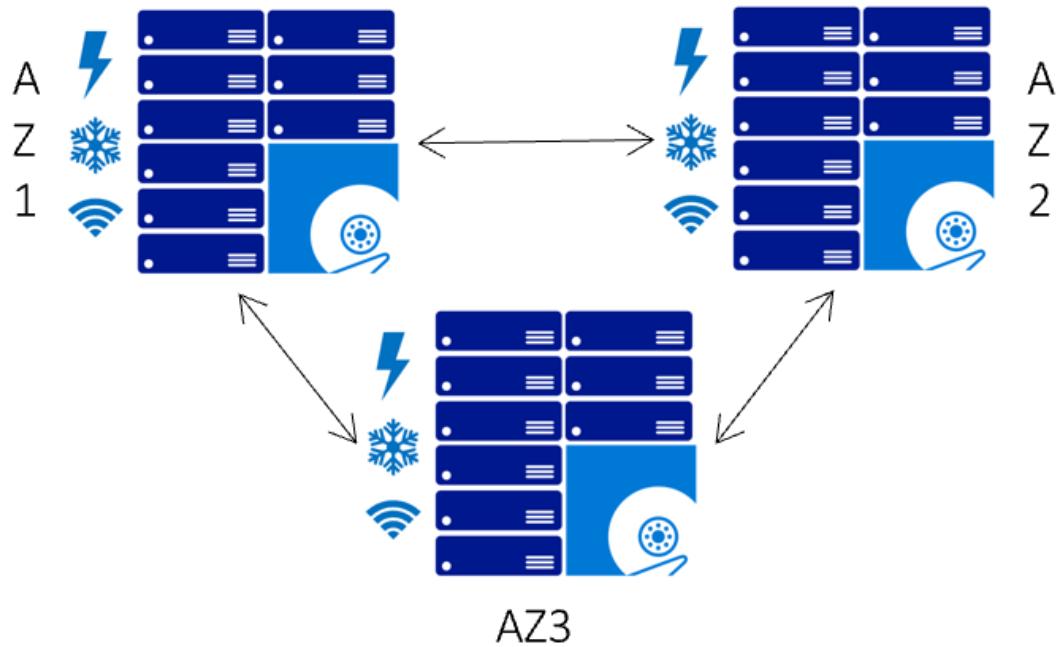
We always recommend that to over-provision the number of replicas due to factors like resource size, content and OS type.



## High availability

[Azure Zone Redundant Storage \(ZRS\)](#) provides resilience against an Availability Zone failure in the region. With the general availability of Azure Compute Gallery, you can choose to store your images in ZRS accounts in regions with Availability Zones.

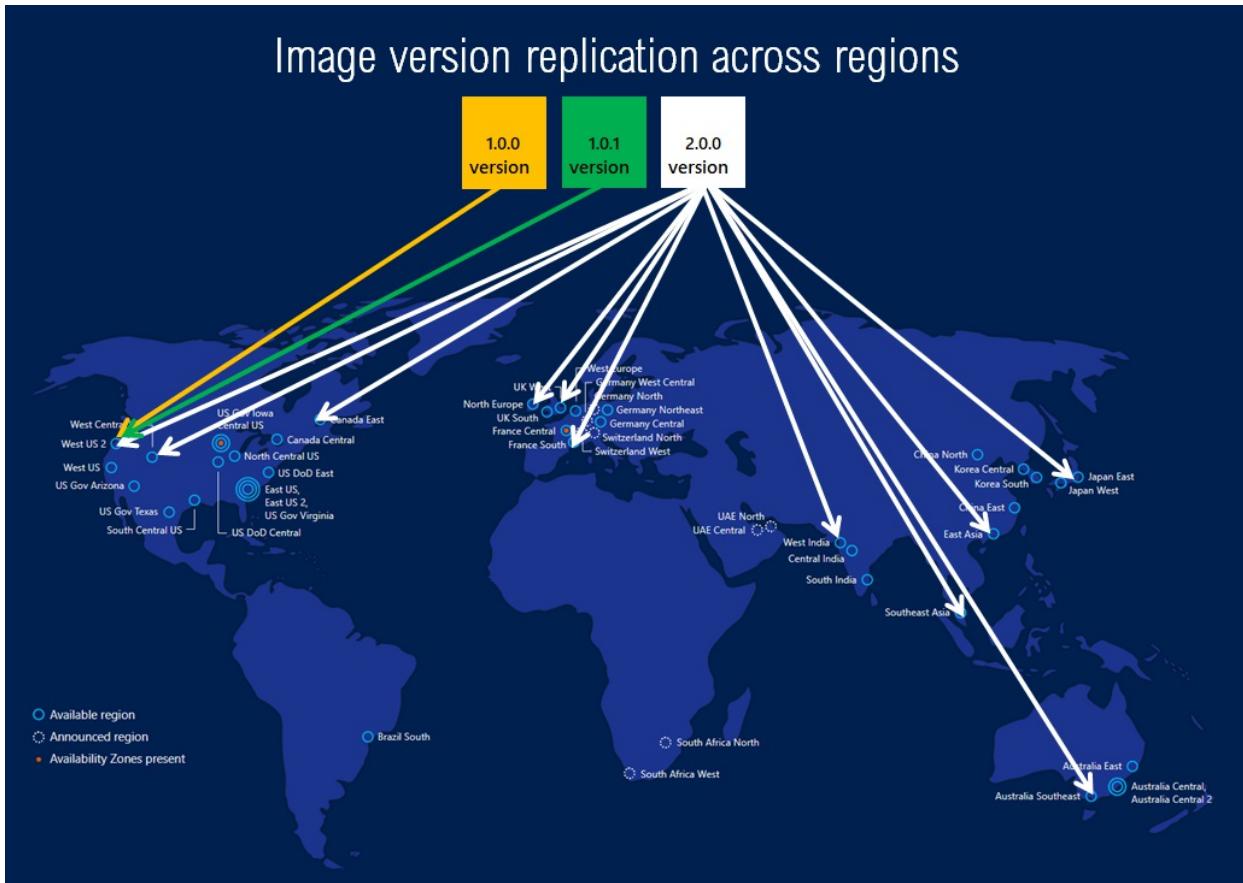
You can also choose the account type for each of the target regions. The default storage account type is Standard\_LRS, but you can choose Standard\_ZRS for regions with Availability Zones. For more information on regional availability of ZRS, see [Data redundancy](#).



## Replication

Azure Compute Gallery also allows you to replicate your resources to other Azure regions automatically. Each image version can be replicated to different regions depending on what makes sense for your organization. One example is to always replicate the latest image in multi-regions while all older image versions are only available in 1 region. This can help save on storage costs.

The regions that a resource is replicated to can be updated after creation time. The time it takes to replicate to different regions depends on the amount of data being copied and the number of regions the version is replicated to. This can take a few hours in some cases. While the replication is happening, you can view the status of replication per region. Once the image replication is complete in a region, you can then deploy a VM or scale-set using that resource in the region.



## Sharing

There are three main ways to share images in an Azure Compute Gallery, depending on who you want to share with:

SHARE WITH:	OPTION
Specific people, groups, or service principals	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
Subscriptions or tenants	Direct shared gallery (preview) lets you share to everyone in a subscription or tenant.
Everyone	Community gallery (preview) lets you share your entire gallery publicly, to all Azure users.

### RBAC

As the Azure Compute Gallery, definition, and version are all resources, they can be shared using the built-in native Azure Roles-based Access Control (RBAC) roles. Using Azure RBAC roles you can share these resources to other users, service principals, and groups. You can even share access to individuals outside of the tenant they were created within. Once a user has access to the resource version, they can use it to deploy a VM or a Virtual Machine Scale Set. Here is the sharing matrix that helps understand what the user gets access to:

SHARED WITH USER	AZURE COMPUTE GALLERY	IMAGE DEFINITION	IMAGE VERSION
Azure Compute Gallery	Yes	Yes	Yes
Image Definition	No	Yes	Yes

We recommend sharing at the Gallery level for the best experience. We do not recommend sharing individual image versions. For more information about Azure RBAC, see [Assign Azure roles](#).

For more information, see [Share using RBAC](#).

### Shared directly to a tenant or subscription

Give specific subscriptions or tenants access to a direct shared Azure Compute Gallery. Sharing a gallery with tenants and subscriptions give them read-only access to your gallery. For more information, see [Share a gallery with subscriptions or tenants](#).

#### IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

To publish images to a direct shared gallery during the preview, you need to register at <https://aka.ms/directsharedgallery-preview>. Creating VMs from a direct shared gallery is open to all Azure users.

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

You can't currently create a Flexible virtual machine scale set from an image shared to you by another tenant.

#### Limitations

During the preview:

- You can only share to subscriptions that are also in the preview.
- You can only share to 30 subscriptions and 5 tenants.
- A direct shared gallery cannot contain encrypted image versions. Encrypted images cannot be created within a gallery that is directly shared.
- Only the owner of a subscription, or a user or service principal assigned to the `Compute Gallery Sharing Admin` role at the subscription or gallery level will be able to enable group-based sharing.
- You need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

### Community gallery

To share a gallery with all Azure users, you can create a community gallery (preview). Community galleries can be used by anyone with an Azure subscription. Someone creating a VM can browse images shared with the community using the portal, REST, or the Azure CLI.

Sharing images to the community is a new capability in [Azure Compute Gallery](#). In the preview, you can make your image galleries public, and share them to all Azure customers. When a gallery is marked as a community gallery, all images under the gallery become available to all Azure customers as a new resource type under `Microsoft.Compute/communityGalleries`. All Azure customers can see the galleries and use them to create VMs. Your original resources of the type `Microsoft.Compute/galleries` are still under your subscription, and private.

For more information, see [Share images using a community gallery](#).

## IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

To publish a community gallery, you need to register for the preview at <https://aka.ms/communitygallery-preview>. Creating VMs from the community gallery is open to all Azure users.

During the preview, the gallery must be created as a community gallery (for CLI, this means using the `--permissions community` parameter) you currently can't migrate a regular gallery to a community gallery.

You can't currently create a Flexible virtual machine scale set from an image shared by another tenant.

## Why share to the community?

As a content publisher, you might want to share a gallery to the community:

- If you have non-commercial, non-proprietary content to share widely on Azure.
- You want greater control over the number of versions, regions, and the duration of image availability.
- You want to quickly share daily or nightly builds with your customers.
- You don't want to deal with the complexity of multi-tenant authentication when sharing with multiple tenants on Azure.

## How sharing with the community works

You [create a gallery resource](#) under `Microsoft.Compute/Galleries` and choose `community` as a sharing option.

When you are ready, you flag your gallery as ready to be shared publicly. Only the owner of a subscription, or a user or service principal with the `Compute Gallery Sharing Admin` role at the subscription or gallery level, can enable a gallery to go public to the community. At this point, the Azure infrastructure creates proxy read-only regional resources, under `Microsoft.Compute/CommunityGalleries`, which are public.

The end-users can only interact with the proxy resources, they never interact with your private resources. As the publisher of the private resource, you should consider the private resource as your handle to the public proxy resources. The `prefix` you provide when you create the gallery will be used, along with a unique GUID, to create the public facing name for your gallery.

Azure users can see the latest image versions shared to the community in the portal, or query for them using the CLI. Only the latest version of an image is listed in the community gallery.

When creating a community gallery, you will need to provide contact information for your images. This information will be shown **publicly**, so be careful when providing it:

- Community gallery prefix
- Publisher support email
- Publisher URL
- Legal agreement URL

Information from your image definitions will also be publicly available, like what you provide for **Publisher**, **Offer**, and **SKU**.

## WARNING

If you want to stop sharing a gallery publicly, you can update the gallery to stop sharing, but making the gallery private will prevent existing virtual machine scale set users from scaling their resources.

If you stop sharing your gallery during the preview, you won't be able to re-share it.

## Limitations for images shared to the community

There are some limitations for sharing your gallery to the community:

- Encrypted images aren't supported.
- For the preview, image resources need to be created in the same region as the gallery. For example, if you create a gallery in West US, the image definitions and image versions should be created in West US if you want to make them available during the public preview.
- For the preview, you can't share [VM Applications](#) to the community.
- The gallery must be created as a community gallery. For the preview, there is no way to migrate an existing gallery to be a community gallery.
- To find images shared to the community from the Azure portal, you need to go through the VM create or scale set creation pages. You can't search the portal or Azure Marketplace for the images.

### IMPORTANT

Microsoft does not provide support for images you share to the community.

## Community-shared images FAQ

### Q: What are the charges for using a gallery that is shared to the community?

A: There are no charges for using the service itself. However, content publishers would be charged for the following:

- Storage charges for application versions and replicas in each of the regions (source and target). These charges are based on the storage account type chosen.
- Network egress charges for replication across regions.

### Q: Is it safe to use images shared to the community?

A: Users should exercise caution while using images from non-verified sources, since these images are not subject to Azure certification.

### Q: If an image that is shared to the community doesn't work, who do I contact for support?

A: Azure is not responsible for any issues users might encounter with community-shared images. The support is provided by the image publisher. Please look up the publisher contact information for the image and reach out to them for any support.

### Q: I have concerns about an image, who do I contact?

A: For issues with images shared to the community:

- To report malicious images, contact [Abuse Report](#).
- To report images that potentially violate intellectual property rights, contact [Infringement Report](#).

### Q: How do I request that an image shared to the community be replicated to a specific region?

A: Only the content publishers have control over the regions their images are available in. If you don't find an image in a specific region, reach out to the publisher directly.

## Activity Log

The [Activity log](#) displays recent activity on the gallery, image, or version including any configuration changes and when it was created and deleted. View the activity log in the Azure portal, or create a [diagnostic setting to send it to a Log Analytics workspace](#), where you can view events over time or analyze them with other collected data.

The following table lists a few example operations that relate to gallery operations in the activity log. For a complete list of possible log entries, see [Microsoft.Compute Resource Provider options](#)

OPERATION	DESCRIPTION
Microsoft.Compute/galleries/write	Creates a new Gallery or updates an existing one
Microsoft.Compute/galleries/delete	Deletes the Gallery
Microsoft.Compute/galleries/share/action	Shares a Gallery to different scopes
Microsoft.Compute/galleries/images/read	Gets the properties of Gallery Image
Microsoft.Compute/galleries/images/write	Creates a new Gallery Image or updates an existing one
Microsoft.Compute/galleries/images/versions/read	Gets the properties of Gallery Image Version

## Billing

There is no extra charge for using the Azure Compute Gallery service. You will be charged for the following resources:

- Storage costs of storing each replica. For images, the storage cost is charged as a snapshot and is based on the occupied size of the image version, the number of replicas of the image version and the number of regions the version is replicated to.
- Network egress charges for replication of the first resource version from the source region to the replicated regions. Subsequent replicas are handled within the region, so there are no additional charges.

For example, let's say you have an image of a 127 GB OS disk, that only occupies 10GB of storage, and one empty 32 GB data disk. The occupied size of each image would only be 10 GB. The image is replicated to 3 regions and each region has two replicas. There will be six total snapshots, each using 10GB. You will be charged the storage cost for each snapshot based on the occupied size of 10 GB. You will pay network egress charges for the first replica to be copied to the additional two regions. For more information on the pricing of snapshots in each region, see [Managed disks pricing](#). For more information on network egress, see [Bandwidth pricing](#).

## Best practices

- To prevent images from being accidentally deleted, use resource locks at the Gallery level. For more information, see [Protect your Azure resources with a lock](#).
- Use ZRS wherever available for high availability. You can configure ZRS in the replication tab when you create the a version of the image or VM application. For more information about which regions support ZRS, see [Azure regions with availability zones](#).
- Keep a minimum of 3 replicas for production images. For every 20 VMs that you create concurrently, we recommend you keep one replica. For example, if you create 1000 VM's concurrently, you should keep 50 replicas (you can have a maximum of 50 replicas per region). To update the replica count, please go to the gallery -> Image Definition -> Image Version -> Update replication.
- Maintain separate galleries for production and test images, don't put them in a single gallery.
- When creating an image definition, keep the Publisher/Offer/SKU consistent with Marketplace images to easily identify OS versions. For example, if you are customizing a Windows server 2019 image from Marketplace and store it as a Compute gallery image, please use the same Publisher/Offer/SKU that is used in the Marketplace image in your compute gallery image.

- Use `excludeFromLatest` when publishing images if you want to exclude a specific image version during VM or scale set creation. [Gallery Image Versions - Create Or Update](#).

If you want to exclude a version in a specific region, use `regionalExcludeFromLatest` instead of the global `excludeFromLatest`. You can set both global and regional `excludeFromLatest` flag, but the regional flag will take precedence when both are specified.

```
"publishingProfile": {
  "targetRegions": [
    {
      "name": "brazilsouth",
      "regionalReplicaCount": 1,
      "regionalExcludeFromLatest": false,
      "storageAccountType": "Standard_LRS"
    },
    {
      "name": "canadacentral",
      "regionalReplicaCount": 1,
      "regionalExcludeFromLatest": true,
      "storageAccountType": "Standard_LRS"
    }
  ],
  "replicaCount": 1,
  "excludeFromLatest": true,
  "storageAccountType": "Standard_LRS"
}
```

- For disaster recovery scenarios, it is a best practice is to have at least two galleries, in different regions. You can still use image versions in other regions, but if the region your gallery is in goes down, you can't create new gallery resources or update existing ones.

## SDK support

The following SDKs support creating Azure Compute Galleries:

- [.NET](#)
- [Java](#)
- [Node.js](#)
- [Python](#)
- [Go](#)

## Templates

You can create Azure Compute Gallery resource using templates. There are several quickstart templates available:

- [Create a gallery](#)
- [Create an image definition in a gallery](#)
- [Create an image version in a gallery](#)

## Next steps

Learn how to deploy [images](#) and [VM apps](#) using an Azure Compute Gallery.

# Create a gallery for storing and sharing resources

9/21/2022 • 7 minutes to read • [Edit Online](#)

An [Azure Compute Gallery](#) (formerly known as Shared Image Gallery) simplifies sharing resources, like images and application packages, across your organization.

The Azure Compute Gallery lets you share custom VM images and application packages with others in your organization, within or across regions, within a tenant. Choose what you want to share, which regions you want to make them available in, and who you want to share them with. You can create multiple galleries so that you can logically group resources.

The gallery is a top-level resource that can be shared in multiple ways:

SHARE WITH:	OPTION
<a href="#">Specific people, groups, or service principals</a>	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
<a href="#">Subscriptions or tenants</a>	Direct shared gallery (preview) lets you share to everyone in a subscription or tenant.
<a href="#">Everyone</a>	Community gallery (preview) lets you share your entire gallery publicly, to all Azure users.

## Naming

Allowed characters for gallery name are uppercase or lowercase letters, digits, dots, and periods. The gallery name can't contain dashes. Gallery names must be unique within your subscription.

## Create a private gallery

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Type **Azure Compute Gallery** in the search box and select **Azure Compute Gallery** in the results.
3. In the Azure Compute Gallery page, click **Add**.
4. On the **Create Azure Compute Gallery** page, select the correct subscription.
5. In **Resource group**, select a resource group from the drop-down or select **Create new** and type a name for the new resource group.
6. In **Name**, type a name for the name of the gallery.
7. Select a **Region** from the drop-down.
8. You can type a short description of the gallery, like *My gallery for testing.* and then click **Review + create**.
9. After validation passes, select **Create**.
10. When the deployment is finished, select **Go to resource**.

# Create a direct shared gallery

## IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

You can't currently create a Flexible virtual machine scale set from an image shared to you by another tenant.

- [Portal](#)
- [CLI](#)
- [REST](#)

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Type **Azure Compute Gallery** in the search box and select **Azure Compute Gallery** in the results.
3. In the **Azure Compute Gallery** page, click **Add**.
4. On the **Create Azure Compute Gallery** page, select the correct subscription.
5. Complete all of the details on the page.
6. At the bottom of the page, select **Next: Sharing method**.

[Home](#) > [Azure compute galleries](#) >

## Create Azure compute gallery

[Basics](#)   [Sharing](#)   [Tags](#)   [Review + create](#)

Azure compute galleries allow you to share images with users or user groups across subscriptions in your organization. Images are published to Azure compute gallery that will be available within Azure Marketplace.

[Learn more about Azure compute galleries](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

myAzureSubscription

Resource group \* ⓘ

(New) myGalleryGroup

[Create new](#)

### Instance details

Name \* ⓘ

myCommunityGallery

Region \* ⓘ

(US) West US

Description ⓘ

Azure Compute Gallery for sharing images publicly.

[Review + create](#)

< Previous

Next : Sharing method >

7. On the **Sharing** tab, select **RBAC + share directly**.

## Create Azure compute gallery

...

Basics    **Sharing**    Tags    Review + create

In addition to role based sharing through Identity Access control, you're able to share the compute gallery using the methods below.

The Azure Compute Gallery - Direct Shared Gallery/Direct shared images (the "PREVIEW") subject to the [Preview Terms for Azure Compute Gallery - Direct Shared Gallery](#).

If you do not agree to these terms, do not use the PREVIEW

Sharing method 

- Role based access control (RBAC)  
Role based sharing through Identity Access control. Share based on permissions assigned to users, groups, and applications at a certain scope. [Learn more](#)
- RBAC + share directly  
Share resources with all users in the same subscription, same tenant, different subscriptions, and different tenants. All users in the subscription or tenant will have read access to the gallery and all the resources within it. [Learn more](#)
- RBAC + share to public community gallery (PREVIEW)  
Publish your Azure compute gallery to the community gallery. Your gallery will be shared with anyone using Azure, including users outside of your organization. [Learn more](#)

[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

8. When you are done, select **Review + create**.

9. After validation passes, select **Create**.

10. When the deployment is finished, select **Go to resource**.

To start sharing the gallery with a subscription or tenant, see [Share a gallery with a subscription or tenant](#).

To start sharing the gallery with a subscription or tenant, use see [Share a gallery with a subscription or tenant](#).

## Create a community gallery

A [community gallery](#) is shared publicly with everyone. To create a community gallery, you create the gallery first, then enable it for sharing. The name of public instance of your gallery will be the prefix you provide, plus a unique GUID.

During the preview, make sure that you create your gallery, image definitions, and image versions in the same region in order to share your gallery publicly.

### IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

To publish a community gallery, you need to register for the preview at <https://aka.ms/communitygallery-preview>. Creating VMs from the community gallery is open to all Azure users.

When creating an image to share with the community, you'll need to provide contact information. This information will be shown **publicly**, so be careful when providing:

- Community gallery prefix
- Publisher support email
- Publisher URL

- Legal agreement URL

Information from your image definitions will also be publicly available, like what you provide for **Publisher**, **Offer**, and **SKU**.

## Prerequisites

Only the owner of a subscription, or a user or service principal assigned to the **Compute Gallery Sharing Admin** role at the subscription or gallery level, can enable a gallery to go public to the community. To assign a role to a user, group, service principal or managed identity, see [Steps to assign an Azure role](#).

- [CLI](#)
- [REST](#)
- [Portal](#)

The `--public-name-prefix` value is used to create a name for the public version of your gallery. The `--public-name-prefix` will be the first part of the public name, and the last part will be a GUID, created by the platform, that is unique to your gallery.

```
location=westus
galleryName=contosoGallery
resourceGroup=myCGRG
publisherUri=https://www.contoso.com
publisherEmail=support@contoso.com
eulaLink=https://www.contoso.com/eula
prefix=ContosoImages

az group create --name $resourceGroup --location $location

az sig create \
  --gallery-name $galleryName \
  --permissions community \
  --resource-group $resourceGroup \
  --publisher-uri $publisherUri \
  --publisher-email $publisherEmail \
  --eula $eulaLink \
  --public-name-prefix $prefix
```

The output of this command will give you the public name for your community gallery in the `sharingProfile` section, under `publicNames`.

To start sharing the gallery to all Azure users, see [Share images using a community gallery](#).

## Next steps

- Create an [image definition and an image version](#).
- [Create a VM application](#) in your gallery.

# Share gallery resources

9/21/2022 • 2 minutes to read • [Edit Online](#)

As the Azure Compute Gallery, definition, and version are all resources, they can be shared using the built-in native Azure Roles-based Access Control (RBAC) roles. Using Azure RBAC roles you can share these resources to other users, service principals, and groups. You can even share access to individuals outside of the tenant they were created within. Once a user has access, they can use the gallery resources to deploy a VM or a Virtual Machine Scale Set. Here's the sharing matrix that helps understand what the user gets access to:

SHARED WITH USER	AZURE COMPUTE GALLERY	IMAGE DEFINITION	IMAGE VERSION
Azure Compute Gallery	Yes	Yes	Yes
Image Definition	No	Yes	Yes

We recommend sharing at the Gallery level for the best experience. We don't recommend sharing individual image versions. For more information about Azure RBAC, see [Assign Azure roles](#).

There are three main ways to share images in an Azure Compute Gallery, depending on who you want to share with:

SHARE WITH:	OPTION
Specific people, groups, or service principals (described in this article)	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
<a href="#">Subscriptions or tenants</a>	A direct shared gallery lets you share to everyone in a subscription or tenant.
<a href="#">Everyone</a>	Community gallery lets you share your entire gallery publicly, to all Azure users.

## Share using RBAC

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. On the page for your gallery, in the menu on the left, select **Access control (IAM)**.
2. Under **Add a role assignment**, select **Add**. The **Add a role assignment** pane will open.
3. Under **Role**, select **Reader**.
4. Under **assign access to**, leave the default of **Azure AD user, group, or service principal**.
5. Under **Select**, type in the email address of the person that you would like to invite.
6. If the user is outside of your organization, you'll see the message **This user will be sent an email that enables them to collaborate with Microsoft**. Select the user with the email address and then click **Save**.

## Next steps

- Create an [image definition](#) and an [image version](#).
- Create a VM from a [generalized](#) or [specialized](#) private gallery.

# Share a gallery with subscriptions or tenants (preview)

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article covers how to share an Azure Compute Gallery with specific subscriptions or tenants using a direct shared gallery. Sharing a gallery with tenants and subscriptions give them read-only access to your gallery.

## IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

To publish images to a direct shared gallery during the preview, you need to register at <https://aka.ms/directsharedgallery-preview>. We will follow up within 5 business days after submitting the form. No additional access required to consume images, Creating VMs from a direct shared gallery is open to all Azure users in the target subscription or tenant the gallery is shared with.

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

There are three main ways to share images in an Azure Compute Gallery, depending on who you want to share with:

SHARE WITH:	OPTION
<a href="#">Specific people, groups, or service principals</a>	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
[Subscriptions or tenants](explained in this article)	Direct shared gallery lets you share to everyone in a subscription or tenant (all users, service principals and managed identities)
<a href="#">Everyone</a>	Community gallery lets you share your entire gallery publicly, to all Azure users.

## Limitations

During the preview:

- You can only share to 30 subscriptions and 5 tenants.
- Only images can be shared. You can't directly share a [VM application](#) during the preview.
- A direct shared gallery can't contain encrypted image versions. Encrypted images can't be created within a gallery that is directly shared.
- Only the owner of a subscription, or a user or service principal assigned to the `Compute Gallery Sharing Admin` role at the subscription or gallery level will be able to enable group-based sharing.
- You need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the

property can't currently be updated.

- TrustedLaunch and ConfidentialVM are not supported
- PowerShell, Ansible, and Terraform aren't supported at this time.
- Not available in Government clouds
- For consuming direct shared images in target subscription, Direct shared images can be found from VM/VMSS creation blade only.
- **Known issue:** When creating a VM from a direct shared image using the Azure portal, if you select a region, select an image, then change the region, you will get an error message: "You can only create VM in the replication regions of this image" even when the image is replicated to that region. To get rid of the error, select a different region, then switch back to the region you want. If the image is available, it should clear the error message.

## Prerequisites

You need to create a [new direct shared gallery](#). A direct shared gallery has the `sharingProfile.permissions` property set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

## How sharing with direct shared gallery works

First you create a gallery under `Microsoft.Compute/Galleries` and choose `groups` as a sharing option.

When you are ready, you share your gallery with subscriptions and tenants. Only the owner of a subscription, or a user or service principal with the `Compute Gallery Sharing Admin` role at the subscription or gallery level, can share the gallery. At this point, the Azure infrastructure creates proxy read-only regional resources, under `Microsoft.Compute/SharedGalleries`. Only subscriptions and tenants you have shared with can interact with the proxy resources, they never interact with your private resources. As the publisher of the private resource, you should consider the private resource as your handle to the public proxy resources. The subscriptions and tenants you have shared your gallery with will see the gallery name as the subscription ID where the gallery was created, followed by the gallery name.

- [Portal](#)
- [CLI](#)
- [REST](#)

### NOTE

**Known issue:** In the Azure portal, If you get an error "Failed to update Azure compute gallery", please verify if you have owner (or) compute gallery sharing admin permission on the gallery.

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Type **Azure Compute Gallery** in the search box and select **Azure Compute Gallery** in the results.
3. In the **Azure Compute Gallery** page, click **Add**.
4. On the **Create Azure Compute Gallery** page, select the correct subscription.
5. Complete all of the details on the page.
6. At the bottom of the page, select **Next: Sharing method**.

## Create Azure compute gallery

...

[Basics](#) [Sharing](#) [Tags](#) [Review + create](#)

Azure compute galleries allow you to share images with users or user groups across subscriptions in your organization. Images are published to Azure compute gallery that will be available within Azure Marketplace.

[Learn more about Azure compute galleries](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

myAzureSubscription

Resource group \* ⓘ

(New) myGalleryGroup

[Create new](#)

### Instance details

Name \* ⓘ

myCommunityGallery

Region \* ⓘ

(US) West US

Description ⓘ

Azure Compute Gallery for sharing images publicly.

[Review + create](#)
[< Previous](#)
[Next : Sharing method >](#)

7. On the **Sharing** tab, select **RBAC + share directly**.

## Create Azure compute gallery

...

[Basics](#) [Sharing](#) [Tags](#) [Review + create](#)

In addition to role based sharing through Identity Access control, you're able to share the compute gallery using the methods below.

The Azure Compute Gallery - Direct Shared Gallery/Direct shared images (the "PREVIEW") subject to the [Preview Terms for Azure Compute Gallery - Direct Shared Gallery](#).

If you do not agree to these terms, do not use the PREVIEW

Sharing method ⓘ

 Role based access control (RBAC)

Role based sharing through Identity Access control. Share based on permissions assigned to users, groups, and applications at a certain scope. [Learn more](#)

 RBAC + share directly

Share resources with all users in the same subscription, same tenant, different subscriptions, and different tenants. All users in the subscription or tenant will have read access to the gallery and all the resources within it. [Learn more](#)

 RBAC + share to public community gallery (PREVIEW)

Publish your Azure compute gallery to the community gallery. Your gallery will be shared with anyone using Azure, including users outside of your organization. [Learn more](#)

[Review + create](#)
[< Previous](#)
[Next : Tags >](#)

8. When you are done, select **Review + create**.

9. After validation passes, select **Create**.

10. When the deployment is finished, select **Go to resource**.

To share the gallery:

1. On the page for the gallery, select **Sharing** from the left menu.

2. Under **Direct sharing settings**, select **Add**.

The screenshot shows the 'Sharing' page with the following interface elements:

- Top navigation: Share, Stop sharing, Refresh.
- Text: In addition to role based sharing through Identity Access control, you're able to share the compute gallery using the methods below.
- Sharing dropdown: RBAC + share directly (PREVIEW).
- Section: Direct sharing settings.
- Buttons: + Add (highlighted with a red box), Delete.
- Inputs: Type (checkbox) and Tenants and subscriptions (dropdown).
- Buttons at the bottom: Save (blue), Cancel.

3. If you would like to share with someone within your organization, for **Type** select *Subscription* or *Tenant* and choose the appropriate item from the **Tenants and subscriptions** drop-down. If you want to share with someone outside of your organization, select either *Subscription outside of my organization* or *Tenant outside of my organization* and then paste or type the ID into the text box.

4. When you are done adding items, select **Save**.

## Next steps

- Create an [image definition and an image version](#).
- Create a VM from a [generalized](#) or [specialized](#) image from a direct shared image in the target subscription or tenant.

# Share images using a community gallery (preview)

9/21/2022 • 5 minutes to read • [Edit Online](#)

To share a gallery with all Azure users, you can create a community gallery (preview). Community galleries can be used by anyone with an Azure subscription. Someone creating a VM can browse images shared with the community using the portal, REST, or the Azure CLI.

Sharing images to the community is a new capability in [Azure Compute Gallery](#). In the preview, you can make your image galleries public, and share them to all Azure customers. When a gallery is marked as a community gallery, all images under the gallery become available to all Azure customers as a new resource type under Microsoft.Compute/communityGalleries. All Azure customers can see the galleries and use them to create VMs. Your original resources of the type `Microsoft.Compute/galleries` are still under your subscription, and private.

## IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

To publish a community gallery, you need to register for the preview at <https://aka.ms/communitygallery-preview>. We will follow up within 5 business days after submitting the form. Creating VMs from the community gallery is open to all Azure users.

During the preview, the gallery must be created as a community gallery (for CLI, this means using the `--permissions community` parameter) you currently can't migrate a regular gallery to a community gallery.

You can't currently create a Flexible virtual machine scale set from an image shared by another tenant.

There are three main ways to share images in an Azure Compute Gallery, depending on who you want to share with:

SHARE WITH:	OPTION
<a href="#">Specific people, groups, or service principals</a>	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
<a href="#">Subscriptions or tenants</a>	Direct shared gallery lets you share to everyone in a subscription or tenant.
<a href="#">Everyone (described in this article)</a>	Community gallery lets you share your entire gallery publicly, to all Azure users.

## Limitations for images shared to the community

There are some limitations for sharing your gallery to the community:

- Encrypted images aren't supported.
- For the preview, image resources need to be created in the same region as the gallery. For example, if you create a gallery in West US, the image definitions and image versions should be created in West US if you want to make them available during the public preview.
- For the preview, you can't share [VM Applications](#) to the community.
- The gallery must be created as a community gallery. For the preview, there is no way to migrate an existing

gallery to be a community gallery.

- To find images shared to the community from the Azure portal, you need to go through the VM create or scale set creation pages. You can't search the portal or Azure Marketplace for the images.

#### IMPORTANT

Microsoft does not provide support for images you share to the community.

## How sharing with the community works

You [create a gallery resource](#) under `Microsoft.Compute/Galleries` and choose `community` as a sharing option.

When you are ready, you flag your gallery as ready to be shared publicly. Only the owner of a subscription, or a user or service principal with the `Compute Gallery Sharing Admin` role at the subscription or gallery level, can enable a gallery to go public to the community. At this point, the Azure infrastructure creates proxy read-only regional resources, under `Microsoft.Compute/CommunityGalleries`, which are public.

The end-users can only interact with the proxy resources, they never interact with your private resources. As the publisher of the private resource, you should consider the private resource as your handle to the public proxy resources. The `prefix` you provide when you create the gallery will be used, along with a unique GUID, to create the public facing name for your gallery.

Azure users can see the latest image versions shared to the community in the portal, or query for them using the CLI. Only the latest version of an image is listed in the community gallery.

When creating a community gallery, you will need to provide contact information for your images. This information will be shown **publicly**, so be careful when providing it:

- Community gallery prefix
- Publisher support email
- Publisher URL
- Legal agreement URL

Information from your image definitions will also be publicly available, like what you provide for **Publisher**, **Offer**, and **SKU**.

#### WARNING

If you want to stop sharing a gallery publicly, you can update the gallery to stop sharing, but making the gallery private will prevent existing virtual machine scale set users from scaling their resources.

If you stop sharing your gallery during the preview, you won't be able to re-share it.

## Start sharing publicly

In order to share a gallery publicly, it needs to be created as a community gallery. For more information, see [Create a community gallery](#)

- [CLI](#)
- [REST](#)
- [Portal](#)

Once you are ready to make the gallery available to the public, enable the community gallery using [az sig share enable-community](#). Only a user in the `Owner` role definition can enable a gallery for community sharing.

```
az sig share enable-community \
--gallery-name $galleryName \
--resource-group $resourceGroup
```

To go back to only RBAC based sharing, use the [az sig share reset](#) command.

To delete a gallery shared to community, you must first run [az sig share reset](#) to stop sharing, then delete the gallery.

#### IMPORTANT

If you are listed as the owner of your subscription, but you are having trouble sharing the gallery publicly, you may need to explicitly [add yourself as owner again](#).

To go back to only RBAC based sharing, use the [az sig share reset](#) command.

To delete a gallery shared to community, you must first run [az sig share reset](#) to stop sharing, then delete the gallery.

## Next steps

Create an [image definition and an image version](#).

Create a VM from a [generalized](#) or [specialized](#) image in a community gallery.

# Store and share images in an Azure Compute Gallery

9/21/2022 • 14 minutes to read • [Edit Online](#)

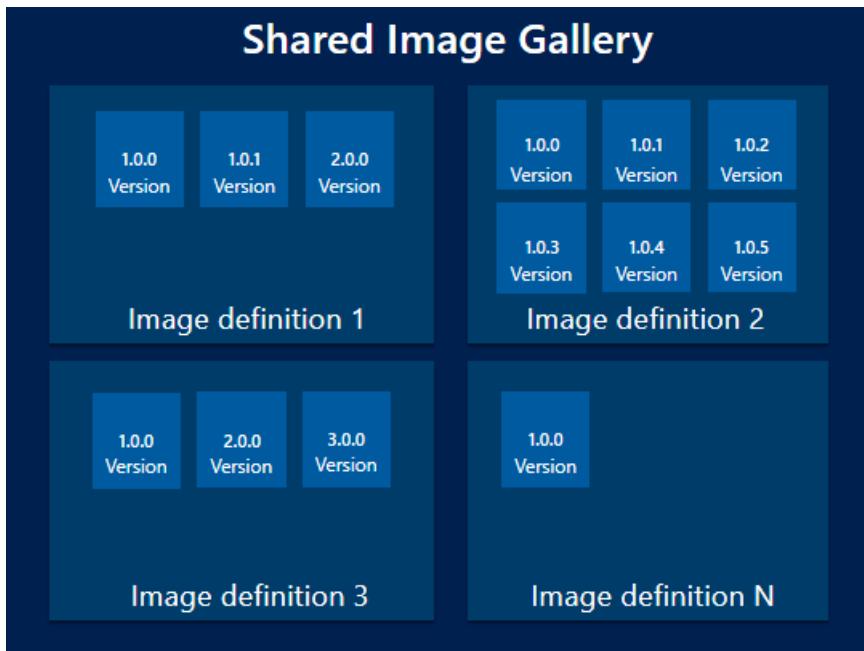
Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

An image is a copy of either a full VM (including any attached data disks) or just the OS disk, depending on how it is created. When you create a VM from the image, a copy of the VHDs in the image are used to create the disks for the new VM. The image remains in storage and can be used over and over again to create new VMs.

If you have a large number of images that you need to maintain, and would like to make them available throughout your company, you can use an [Azure Compute Gallery](#) as a repository.

When you use a gallery to store images, multiple resource types are created:

RESOURCE	DESCRIPTION
<b>Image source</b>	This is a resource that can be used to create an <b>image version</b> in a gallery. An image source can be an existing Azure VM that is either <a href="#">generalized or specialized</a> , a managed image, a snapshot, a VHD or an image version in another gallery.
<b>Gallery</b>	Like the Azure Marketplace, a <b>gallery</b> is a repository for managing and sharing images and other resources, but you control who has access.
<b>Image definition</b>	Image definitions are created within a gallery and they carry information about the image and any requirements for using it to create VMs. This includes whether the image is Windows or Linux, release notes, and minimum and maximum memory requirements. It is a definition of a type of image.
<b>Image version</b>	An <b>image version</b> is what you use to create a VM when using a gallery. You can have multiple versions of an image as needed for your environment. Like a managed image, when you use an <b>image version</b> to create a VM, the image version is used to create new disks for the VM. Image versions can be used multiple times.



## Image definitions

Image definitions are a logical grouping for versions of an image. The image definition holds information about why the image was created, what OS it is for, and other information about using the image. An image definition is like a plan for all of the details around creating a specific image. You don't deploy a VM from an image definition, but from the image versions created from the definition.

There are three parameters for each image definition that are used in combination - **Publisher**, **Offer** and **SKU**. These are used to find a specific image definition. You can have image versions that share one or two, but not all three values. For example, here are three image definitions and their values:

IMAGE DEFINITION	PUBLISHER	OFFER	SKU
myImage1	Contoso	Finance	Backend
myImage2	Contoso	Finance	Frontend
myImage3	Testing	Finance	Frontend

All three of these have unique sets of values. The format is similar to how you can currently specify publisher, offer, and SKU for [Azure Marketplace images](#) in Azure PowerShell to get the latest version of a Marketplace image. Each image definition needs to have a unique set of these values.

The following parameters determine which types of image versions they can contain:

- Operating system state - You can set the OS state to [generalized or specialized](#). This field is required.
- Operating system - can be either Windows or Linux. This field is required.
- Hyper-V generation - specify whether the image was created from a generation 1 or [generation 2](#) Hyper-V VHD. Default is generation 1.

The following are other parameters that can be set on your image definition so that you can more easily track your resources:

- Description - use description to give more detailed information on why the image definition exists. For example, you might have an image definition for your front-end server that has the application pre-installed.
- EULA - can be used to point to an end-user license agreement specific to the image definition.
- Privacy Statement and Release notes - store release notes and privacy statements in Azure storage and

provide a URI for accessing them as part of the image definition.

- End-of-life date - establish a default date after which the image shouldn't be used, for all image versions in the image definition. End-of-life dates are informational; users will still be able to create VMs from images and versions past the end-of-life date.
- Tag - you can add tags when you create your image definition. For more information about tags, see [Using tags to organize your resources](#)
- Minimum and maximum vCPU and memory recommendations - if your image has vCPU and memory recommendations, you can attach that information to your image definition.
- Disallowed disk types - you can provide information about the storage needs for your VM. For example, if the image isn't suited for standard HDD disks, you add them to the disallow list.
- Purchase plan information for Marketplace images - `-PurchasePlanPublisher`, `-PurchasePlanName`, and `-PurchasePlanProduct`. For more information about purchase plan information, see [Find images in the Azure Marketplace](#) and [Supply Azure Marketplace purchase plan information when creating images](#).

## Image versions

An **image version** is what you use to create a VM. You can have multiple versions of an image as needed for your environment. When you use an **image version** to create a VM, the image version is used to create new disks for the VM. Image versions can be used multiple times.

The properties of an image version are:

- Version number. This is used as the name of the image version. It is always in the format: MajorVersion.MinorVersion.Patch. When you specify to use **latest** when creating a VM, the latest image is chosen based on the highest MajorVersion, then MinorVersion, then Patch.
- Source. The source can be a VM, managed disk, snapshot, managed image, or another image version.
- Exclude from latest. You can keep a version from being used as the latest image version.
- End of life date. Indicate the end-of-life date for the image version. End-of-life dates are informational; users will still be able to create VMs from versions past the end-of-life date.

## Generalized and specialized images

There are two operating system states supported by Azure Compute Gallery. Typically images require that the VM used to create the image has been **generalized** before taking the image. Generalizing is a process that removes machine and user specific information from the VM. For Linux, you can use `waagent -deprovision` or `-deprovision+user` parameters. For Windows, the Sysprep tool is used.

Specialized VMs have not been through a process to remove machine specific information and accounts. Also, VMs created from specialized images do not have an `osProfile` associated with them. This means that specialized images will have some limitations in addition to some benefits.

- VMs and scale sets created from specialized images can be up and running quicker. Because they are created from a source that has already been through first boot, VMs created from these images boot faster.
- Accounts that could be used to log into the VM can also be used on any VM created using the specialized image that is created from that VM.
- VMs will have the **Computer name** of the VM the image was taken from. You should change the computer name to avoid collisions.
- The `osProfile` is how some sensitive information is passed to the VM, using `secrets`. This may cause issues using KeyVault, WinRM and other functionality that uses `secrets` in the `osProfile`. In some cases, you can use managed service identities (MSI) to work around these limitations.

## Updating resources

Once created, you can make some changes to the gallery resources. These are limited to:

Azure Compute Gallery:

- Description

Image definition:

- Recommended vCPUs
- Recommended memory
- Description
- End of life date

Image version:

- Regional replica count
- Target regions
- Exclude from latest
- End of life date

## Sharing

There are three main ways to share an Azure Compute Gallery, depending on who you want to share with:

SHARE WITH:	OPTION
Specific people, groups, or service principals	Role-based access control (RBAC) lets you share resources to specific people, groups, or service principals on a granular level.
Subscriptions or tenants	A direct shared gallery (preview) lets you share to everyone in a subscription or tenant.
Everyone	Community gallery (preview) lets you share your entire gallery publicly, to all Azure users.

## Shallow replication

When you create an image version, you can set the replication mode to shallow for development and test.

Shallow replication skips copying the image, so the image version is ready much faster. But, it also means you can't deploy a large number of VMs from that image version. This is similar to the way that the older managed images worked.

Shallow replication can also be useful if you have very large images (up to 32TB) that aren't frequently deployed. Because the source image isn't copied, larger disks can be used. But, they also can't be used for deploying large numbers of VMs concurrently.

To set an image for shallow replication, use `--replication-mode Shallow` with the Azure CLI.

## SDK support

The following SDKs support creating Azure Compute Galleries:

- [.NET](#)
- [Java](#)
- [Node.js](#)

- [Python](#)
- [Go](#)

## Templates

You can create Azure Compute Gallery resource using templates. There are several quickstart templates available:

- [Create a gallery](#)
- [Create an image definition in a gallery](#)
- [Create an image version in a gallery](#)

## Frequently asked questions

- [How can I list all the Azure Compute Gallery resources across subscriptions?](#)
- [Can I move my existing image to an Azure Compute Gallery?](#)
- [Can I create an image version from a specialized disk?](#)
- [Can I move the Azure Compute Gallery resource to a different subscription after it has been created?](#)
- [Can I replicate my image versions across clouds such as Azure China 21Vianet, Azure Germany, or Azure Government Cloud?](#)
- [Can I replicate my image versions across subscriptions?](#)
- [Can I share image versions across Azure AD tenants?](#)
- [How long does it take to replicate image versions across the target regions?](#)
- [What is the difference between source region and target region?](#)
- [How do I specify the source region while creating the image version?](#)
- [How do I specify the number of image version replicas to be created in each region?](#)
- [Can I create the gallery in a different location than the one for the image definition and image version?](#)
- [What are the charges for using an Azure Compute Gallery?](#)
- [What API version should I use when creating images?](#)
- [What API version should I use to create a VM or Virtual Machine Scale Set out of the image version?](#)
- [Can I update my Virtual Machine Scale Set created using managed image to use Azure Compute Gallery images?](#)

### How can I list all the Azure Compute Gallery resources across subscriptions?

To list all the Azure Compute Gallery resources across subscriptions that you have access to on the Azure portal, follow the steps below:

1. Open the [Azure portal](#).
  2. Scroll down the page and select **All resources**.
  3. Select all the subscriptions under which you'd like to list all the resources.
  4. Look for resources of the **Azure Compute Gallery** type.
- [Azure CLI](#)
  - [Azure PowerShell](#)

To list all the Azure Compute Gallery resources, across subscriptions that you have permissions to, use the following command in the Azure CLI:

```
az account list -otsv --query "[].id" | xargs -n 1 az sig list --subscription
```

For more information, see [List, update, and delete image resources](#).

## **Can I move my existing image to an Azure Compute Gallery?**

Yes. There are 3 scenarios based on the types of images you may have.

Scenario 1: If you have a managed image, then you can create an image definition and image version from it. For more information, see [Create and image definition and an image version](#).

Scenario 2: If you have an unmanaged image, you can create a managed image from it, and then create an image definition and image version from it.

Scenario 3: If you have a VHD in your local file system, then you need to upload the VHD to a managed image, then you can create an image definition and image version from it.

- If the VHD is of a Windows VM, see [Upload a VHD](#).
- If the VHD is for a Linux VM, see [Upload a VHD](#)

## **Can I create an image version from a specialized disk?**

Yes, you can create a VM from a [specialized image](#).

## **Can I move the Azure Compute Gallery resource to a different subscription after it has been created?**

No, you can't move the gallery image resource to a different subscription. You can replicate the image versions in the gallery to other regions or copy an [image from another gallery](#).

## **Can I replicate my image versions across clouds such as Azure China 21Vianet or Azure Germany or Azure Government Cloud?**

No, you cannot replicate image versions across clouds.

## **Can I replicate my image versions across subscriptions?**

No, you may replicate the image versions across regions in a subscription and use it in other subscriptions through RBAC.

## **Can I share image versions across Azure AD tenants?**

Yes, you can use RBAC to share to individuals across tenants. But, to share at scale, see "Share gallery images across Azure tenants" using [PowerShell](#) or [CLI](#).

## **How long does it take to replicate image versions across the target regions?**

The image version replication time is entirely dependent on the size of the image and the number of regions it is being replicated to. However, as a best practice, it is recommended that you keep the image small, and the source and target regions close for best results. You can check the status of the replication using the -ReplicationStatus flag.

## **What is the difference between source region and target region?**

Source region is the region in which your image version will be created, and target regions are the regions in which a copy of your image version will be stored. For each image version, you can only have one source region. Also, make sure that you pass the source region location as one of the target regions when you create an image version.

## **How do I specify the source region while creating the image version?**

While creating an image version, you can use the `--location` argument in CLI and the `-Location` parameter in PowerShell to specify the source region. Please ensure the managed image that you are using as the base image to create the image version is in the same location as the location in which you intend to create the image version. Also, make sure that you pass the source region location as one of the target regions when you create an image version.

## **How do I specify the number of image version replicas to be created in each region?**

There are two ways you can specify the number of image version replicas to be created in each region:

1. The regional replica count which specifies the number of replicas you want to create per region.
  2. The common replica count which is the default per region count in case regional replica count is not specified.
- [Azure CLI](#)
  - [Azure PowerShell](#)

To specify the regional replica count, pass the location along with the number of replicas you want to create in that region: "South Central US=2".

If regional replica count is not specified with each location, then the default number of replicas will be the common replica count that you specified.

To specify the common replica count in Azure CLI, use the `--replica-count` argument in the `az sig image-version create` command.

#### **Can I create the gallery in a different location than the one for the image definition and image version?**

Yes, it is possible. But, as a best practice, we encourage you to keep the resource group, gallery, image definition, and image version in the same location.

#### **What are the charges for using an Azure Compute Gallery?**

There are no charges for using an Azure Compute Gallery, except the storage charges for storing the image versions and network egress charges for replicating the image versions from source region to target regions.

#### **What API version should I use when creating images?**

To work with galleries, image definitions, and image versions, we recommend you use API version 2018-06-01. Zone Redundant Storage (ZRS) requires version 2019-03-01 or later.

#### **What API version should I use to create a VM or Virtual Machine Scale Set out of the image version?**

For VM and Virtual Machine Scale Set deployments using an image version, we recommend you use API version 2018-04-01 or higher.

#### **Can I update my Virtual Machine Scale Set created using managed image to use Azure Compute Gallery images?**

Yes, you can update the scale set image reference from a managed image to an Azure Compute Gallery image, as long as the OS type, Hyper-V generation, and the data disk layout matches between the images.

## Troubleshoot

If you have issues with performing any operations on the gallery resources, consult the list of common errors in the [troubleshooting guide](#).

In addition, you can post and tag your question with `azure-virtual-machines-images` at [Q&A](#).

## Next steps

Learn how to deploy images using the [Azure Compute Gallery](#).

# Create an image of a VM in the portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

An image can be created from a VM and then used to create multiple VMs.

For images stored in an Azure Compute Gallery (formerly known as Shared Image Gallery), you can use VMs that already have accounts created on them (specialized) or you can generalize the VM before creating the image to remove machine accounts and other machine specific information. To generalize a VM, see [Generalized a Windows VM](#). For more information, see [Generalized and specialized images](#).

## Capture a VM in the portal

1. Go to the [Azure portal](#), then search for and select **Virtual machines**.
2. Select your VM from the list.
3. On the page for the VM, on the upper menu, select **Capture**.

The **Create an image** page appears.

4. For **Resource group**, either select **Create new** and enter a name, or select a resource group to use from the drop-down list. If you want to use an existing gallery, select the resource group for the gallery you want to use.
5. To create the image in a gallery, select **Yes, share it to a gallery as an image version**.

To only create a managed image, select **No, capture only a managed image**. The VM must have been generalized to create a managed image. The only other required information is a name for the image.

6. If you want to delete the source VM after the image has been created, select **Automatically delete this virtual machine after creating the image**. This is not recommended.
7. For **Gallery details**, select the gallery or create a new gallery by selecting **Create new**.
8. In **Operating system state** select generalized or specialized. For more information, see [Generalized and specialized images](#).
9. Select an image definition or select **create new** and provide a name and information for a new [Image definition](#).
10. Enter an [image version](#) number. If this is the first version of this image, type **1.0.0**.
11. If you want this version to be included when you specify **latest** for the image version, then leave **Exclude from latest** unchecked.
12. Select an **End of life** date. This date can be used to track when older images need to be retired.
13. Under **Replication**, select a default replica count and then select any additional regions where you would like your image replicated.
14. When you are done, select **Review + create**.
15. After validation passes, select **Create** to create the image.

## Next steps

- Azure Compute Galleries overview

# Create an image definition and an image version

9/21/2022 • 12 minutes to read • [Edit Online](#)

A [Azure Compute Gallery](#) (formerly known as Shared Image Gallery) simplifies custom image sharing across your organization. Custom images are like marketplace images, but you create them yourself. Images can be created from a VM, VHD, snapshot, managed image, or another image version.

The Azure Compute Gallery lets you share your custom VM images with others in your organization, within or across regions, within an Azure AD tenant, or publicly using a [community gallery \(preview\)](#). Choose which images you want to share, which regions you want to make them available in, and who you want to share them with. You can create multiple galleries so that you can logically group images.

The Azure Compute Gallery feature has multiple resource types:

RESOURCE	DESCRIPTION
Image source	This is a resource that can be used to create an <b>image version</b> in a gallery. An image source can be an existing Azure VM that is either <a href="#">generalized or specialized</a> , a managed image, a snapshot, or an image version in another gallery.
Gallery	Like the Azure Marketplace, a <b>gallery</b> is a repository for managing and sharing images and <a href="#">VM applications</a> , but you control who has access.
Image definition	Image definitions are created within a gallery and carry information about the image and requirements for using it internally. This includes whether the image is Windows or Linux, release notes, and minimum and maximum memory requirements. It is a definition of a type of image.
Image version	An <b>image version</b> is what you use to create a VM when using a gallery. You can have multiple versions of an image as needed for your environment. Like a managed image, when you use an <b>image version</b> to create a VM, the image version is used to create new disks for the VM. Image versions can be used multiple times.

## Before you begin

To complete this article, you must have an existing Azure Compute Gallery, and a source for your image available in Azure. Image sources can be:

- A VM in your subscription. You can capture an image from both [specialized and generalized](#) VMs.
- A Managed image,
- Managed OS and data disks.
- OS and data disks as VHDs in a storage account.
- Other image versions either in the same gallery or another gallery in the same subscription.

If the image will contain data disks, the data disk size cannot be more than 1 TB.

Image definition names can be made up of uppercase or lowercase letters, digits, dots, dashes and periods. For

more information about the values you can specify for an image definition, see [Image definitions](#).

Allowed characters for the image version are numbers and periods. Numbers must be within the range of a 32-bit integer. Format: *MajorVersion.MinorVersion.Patch*.

When working through this article, replace the resource names where needed.

For [generalized](#) images, see the OS specific guidance before capturing the image:

- **Linux**

- [Generic steps](#)
- [CentOS](#)
- [Debian](#)
- [Flatcar](#)
- [FreeBSD](#)
- [Oracle Linux](#)
- [OpenBSD](#)
- [Red Hat](#)
- [SUSE](#)
- [Ubuntu](#)

- **Windows**

If you plan to run Sysprep before uploading your virtual hard disk (VHD) to Azure for the first time, make sure you have [prepared your VM](#).

## Community gallery (preview)

### IMPORTANT

Azure Compute Gallery – community gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

To share images in the community gallery, you need to register for the preview at <https://aka.ms/communitygallery-preview>. Creating VMs and scale sets from images shared the community gallery is open to all Azure users.

Information from your image definitions will be publicly available, like what you provide for **Publish**, **Offer**, and **SKU**.

If you will be sharing your images using a [community gallery \(preview\)](#), make sure that you create your gallery, image definitions, and image versions in the same region.

When users search for community gallery images, only the latest version of an image is shown.

## Create an image

Choose an option below for creating your image definition and image version:

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

To create an image from a VM in the portal, see [Capture an image of a VM](#).

To create an image using a source other than a VM, follow these steps.

1. Go to the [Azure portal](#), then search for and select **Azure Compute Gallery**.
2. Select the gallery you want to use from the list.
3. On the page for your gallery, select **Add** from the top of the page and then select **VM image definition** from the drop-down.
4. on the **Add new image definition to Azure Compute Gallery** page, in the **Basics** tab, select a **Region**.
5. For **Image definition name**, type a name like *myImageDefinition*.
6. For **Operating system**, select the correct option based on your source.
7. For **VM generation**, select the option based on your source. In most cases, this will be *Gen 1*. For more information, see [Support for generation 2 VMs](#).
8. For **Operating system state**, select the option based on your source. For more information, see [Generalized and specialized](#).
9. For **Publisher**, type a unique name like *myPublisher*.
10. For **Offer**, type a unique name like *myOffer*.
11. For **SKU**, type a unique name like *mySKU*.
12. At the bottom of the page, select **Review + create**.
13. After the image definition passes validation, select **Create**.
14. When the deployment is finished, select **Go to resource**.
15. In the page for your image definition, on the **Get started** tab, select **Create a version**.
16. In **Region**, select the region where you want the image created. In some cases, the source must be in the same region where the image is created. If you aren't seeing your source listed in later drop-downs, try changing the region for the image. You can always replicate the image to other regions later.
17. For **Version number**, type a number like *1.0.0*. The image version name should follow *major.minor.patch* format using integers.
18. In **Source**, select the type of file you are using for your source from the drop-down. See the table below for specific details for each source type.

SOURCE	OTHER FIELDS
Disks or snapshots	<ul style="list-style-type: none"> <li>- For <b>OS disk</b> select the disk or snapshot from the drop-down.</li> <li>- To add a data disk, type the LUN number and then select the data disk from the drop-down.</li> </ul>
Image version	<ul style="list-style-type: none"> <li>- Select the <b>Source gallery</b> from the drop-down.</li> <li>- Select the correct image definition from the drop-down.</li> <li>- Select the existing image version that you want to use from the drop-down.</li> </ul>
Managed image	Select the <b>Source image</b> from the drop-down. The managed image must be in the same region that you chose in <b>Instance details</b> .

SOURCE	OTHER FIELDS
VHD in a storage account	Select <b>Browse</b> to choose the storage account for the VHD.

19. In **Exclude from latest**, leave the default value of *No* unless you don't want this version used when creating a VM using `latest` instead of a version number.
20. For **End of life date**, select a date from the calendar for when you think this version should stop being used.
21. In the **Replication** tab, select the storage type from the drop-down.
22. Set the **Default replica count**, you can override this for each region you add.
23. You need to replicate to the source region, so the first replica in the list will be in the region where you created the image. You can add more replicas by selecting the region from the drop-down and adjusting the replica count as necessary.
24. When you are done, select **Review + create**. Azure will validate the configuration.
25. When image version passes validation, select **Create**.
26. When the deployment is finished, select **Go to resource**.

It can take a while to replicate the image to all of the target regions.

You can also capture an existing VM as an image, from the portal. For more information, see [Create an image of a VM in the portal](#).

## New Features

Many new features like ARM64, Accelerated Networking, TrustedVMSupported etc. are only supported through Azure Compute Gallery and not available for 'Managed images'. For a complete list of new features available through Azure Compute Gallery, please refer <https://learn.microsoft.com/cli/azure/sig/image-version?view=azure-cli-latest#az-sig-image-version-create>

## Next steps

For information about how to supply purchase plan information, see [Supply Azure Marketplace purchase plan information when creating images](#).

# Create a VM from a generalized image version

9/21/2022 • 13 minutes to read • [Edit Online](#)

Create a VM from a [generalized image version](#) stored in an Azure Compute Gallery (formerly known as Shared Image Gallery). If you want to create a VM using a specialized image, see [Create a VM from a specialized image](#).

This article shows how to create a VM from a generalized image:

- [In your own gallery](#)
- Shared to a [community gallery](#)
- [Directly shared to your subscription or tenant](#)

## Create a VM from your gallery

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

Now you can create one or more new VMs. This example creates a VM named *myVM*, in the *myResourceGroup*, in the *East US* datacenter.

1. Go to your image definition. You can use the resource filter to show all image definitions available.
2. On the page for your image definition, select **Create VM** from the menu at the top of the page.
3. For **Resource group**, select **Create new** and type *myResourceGroup* for the name.
4. In **Virtual machine name**, type *myVM*.
5. For **Region**, select *East US*.
6. For **Availability options**, leave the default of *No infrastructure redundancy required*.
7. The value for **Image** is automatically filled with the **latest** image version if you started from the page for the image definition.
8. For **Size**, choose a VM size from the list of available sizes and then choose **Select**.
9. Under **Administrator account**, you need to provide a username, such as *azureuser* and a password or SSH key. The password must be at least 12 characters long and meet the [defined complexity requirements](#).
10. If you want to allow remote access to the VM, under **Public inbound ports**, choose **Allow selected ports** and then select **SSH (22)** or **RDP (3389)** from the drop-down. If you don't want to allow remote access to the VM, leave **None** selected for **Public inbound ports**.
11. When you are finished, select the **Review + create** button at the bottom of the page.
12. After the VM passes validation, select **Create** at the bottom of the page to start the deployment.

## Create a VM from a community gallery image

### IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

Microsoft does not provide support for images in the [community gallery](#).

- [CLI](#)
- [Portal](#)
- [REST](#)

To create a VM using an image shared to a community gallery, use the unique ID of the image for the `--image` which will be in the following format:

```
/CommunityGalleries/<community gallery name, like: ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f>/Images/<image name>/Versions/latest
```

As an end user, to get the public name of a community gallery, you need to use the portal. Go to **Virtual machines > Create > Azure virtual machine > Image > See all images > Community Images > Public gallery name.**

In this example, we are creating a VM from a Linux image and creating SSH keys for authentication.

```
imgDef="/CommunityGalleries/ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f/Images/myLinuxImage/Versions/latest"
vmResourceGroup=myResourceGroup
location=eastus
vmName=myVM
adminUsername=azureuser

az group create --name $vmResourceGroup --location $location

az vm create \
--resource-group $vmResourceGroup \
--name $vmName \
--image $imgDef \
--admin-username $adminUsername \
--generate-ssh-keys
```

When using a community image, you'll be prompted to accept the legal terms. The message will look like this:

To create the VM from community gallery image, you must accept the license agreement and privacy statement:  
<http://contoso.com>. (If you want to accept the legal terms by default, please use the option '--accept-term' when creating VM/VMSS) (Y/n):

## Create a VM from a gallery shared with your subscription or tenant

### IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

To publish images to a direct shared gallery during the preview, you need to register at <https://aka.ms/directsharedgallery-preview>. Creating VMs from a direct shared gallery is open to all Azure users.

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

- [CLI](#)
- [Portal](#)
- [REST](#)

To create a VM using an image shared to your subscription or tenant, you need the unique ID of the image in the following format:

```
/SharedGalleries/<uniqueID>/Images/<image name>/Versions/latest
```

To find the `uniqueID` of a gallery that is shared with you, use [az sig list-shared](#). In this example, we are looking for galleries in the West US region.

```
region=westus
az sig list-shared --location $region --query "[].name" -o tsv
```

Use the gallery name to find the images that are available. In this example, we list all of the images in *West US* and by name, the unique ID that is needed to create a VM, OS and OS state.

```
galleryName="1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f-myDirectShared"
az sig image-definition list-shared \
--gallery-unique-name $galleryName \
--location $region \
--query [*].{"Name:name,ID:uniqueId,OS:osType,State:osState}" -o table
```

Make sure the state of the image is `Generalized`. If you want to use an image with the `Specialized` state, see [Create a VM from a specialized image version](#).

Use the `Id` from the output, appended with `/Versions/latest` to use the latest version, as the value for `--image` to create a VM. In this example, we are creating a VM from a Linux image that is directly shared to us, and creating SSH keys for authentication.

```
imgDef="/SharedGalleries/1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f-
MYDIRECTSHARED/Images/myDirectDefinition/Versions/latest"
vmResourceGroup=myResourceGroup
location=westus
vmName=myVM
adminUsername=azureuser

az group create --name $vmResourceGroup --location $location

az vm create\
--resource-group $vmResourceGroup \
--name $vmName \
--image $imgDef \
--admin-username $adminUsername \
--generate-ssh-keys
```

## Next steps

- [Create an Azure Compute Gallery](#)
- [Create an image in an Azure Compute Gallery](#)

# Create a VM using a specialized image version

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

Create a VM from a [specialized image version](#) stored in an Azure Compute Gallery (formerly known as Shared Image Gallery). If you want to create a VM using a generalized image version, see [Create a VM from a generalized image version](#).

This article shows how to create a VM from a specialized image:

- [In your own gallery](#)
- Shared to a [community gallery](#)
- [Directly shared to your subscription or tenant](#)

## IMPORTANT

When you create a new VM from a specialized image, the new VM retains the computer name of the original VM. Other computer-specific information, like the CMID, is also kept. This duplicate information can cause issues. When copying a VM, be aware of what types of computer-specific information your applications rely on.

## Create a VM from your gallery

- [CLI](#)
- [PowerShell](#)
- [Portal](#)

List the image definitions in a gallery using [az sig image-definition list](#) to see the name and ID of the definitions.

```
resourceGroup=myGalleryRG
gallery=myGallery
az sig image-definition list \
--resource-group $resourceGroup \
--gallery-name $gallery \
--query "[].[name, id]" \
--output tsv
```

Create the VM using [az vm create](#) using the --specialized parameter to indicate that the image is a specialized image.

Use the image definition ID for `--image` to create the VM from the latest version of the image that is available. You can also create the VM from a specific version by supplying the image version ID for `--image`.

In this example, we're creating a VM from the latest version of the *myImageDefinition* image.

```
az group create --name myResourceGroup --location eastus
az vm create --resource-group myResourceGroup \
--name myVM \
--image "/subscriptions/<Subscription
ID>/resourceGroups/myGalleryRG/providers/Microsoft.Compute/galleries/myGallery/images/myImageDefinition" \
--specialized
```

# Create a VM from a community gallery image

## IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

Microsoft does not provide support for images in the [community gallery](#).

- [CLI](#)
- [Portal](#)

To create a VM using an image shared to a community gallery, use the unique ID of the image for the `--image`, which will be in the following format:

```
/CommunityGalleries/<community gallery name, like: ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f>/Images/<image name>/Versions/latest
```

As an end user, to get the public name of a community gallery, you need to use the portal. Go to **Virtual machines > Create > Azure virtual machine > Image > See all images > Community Images > Public gallery name**.

List all of the image definitions that are available in a community gallery using [az sig image-definition list-community](#). In this example, we list all of the images in the *ContosoImage* gallery in *West US* and by name, the unique ID that is needed to create a VM, OS and OS state.

```
az sig image-definition list-community \
--public-gallery-name "ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f" \
--location westus \
--query [*].{"Name:name, ID:uniqueId, OS:osType, State:osState}" -o table
```

To create a VM from a generalized image in a community gallery, see [Create a VM from a generalized image version](#).

Create the VM using `az vm create` using the `--specialized` parameter to indicate that the image is a specialized image.

In this example, we're creating a VM from the latest version of the *myImageDefinition* image.

```
az group create --name myResourceGroup --location eastus
az vm create --resource-group myResourceGroup \
--name myVM \
--image "/CommunityGalleries/ContosoImages-f61bb1d9-3c5a-4ad2-99b5-744030225de6/Images/LinuxSpecializedVersions/latest" \
--specialized
```

When using a community image, you'll be prompted to accept the legal terms. The message will look like this:

```
To create the VM from community gallery image, you must accept the license agreement and privacy statement:
http://contoso.com. (If you want to accept the legal terms by default, please use the option '--accept-term' when creating VM/VMSS) (Y/n):
```

## Create a VM from a gallery shared with your subscription or tenant

## IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

To publish images to a direct shared gallery during the preview, you need to register at <https://aka.ms/directsharedgallery-preview>. Creating VMs from a direct shared gallery is open to all Azure users.

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

- [CLI](#)
- [Portal](#)

To create a VM using the latest version of an image shared to your subscription or tenant, you need the ID of the image in the following format:

```
/SharedGalleries/<uniqueID>/Images/<image name>/Versions/latest
```

To find the `uniqueID` of a gallery that is shared with you, use [az sig list-shared](#). In this example, we're looking for galleries in the West US region.

```
region=westus
az sig list-shared --location $region --query "[]?.name" -o tsv
```

Use the gallery name to find all of the images that are available. In this example, we list all of the images in *West US* and by name, the unique ID that is needed to create a VM, OS and OS state.

```
galleryName="1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f-myDirectShared"
az sig image-definition list-shared \
--gallery-unique-name $galleryName \
--location $region \
--query "[*].'{Name:name, ID:uniqueId, OS:osType, State:osState}" -o table
```

Make sure the state of the image is `Specialized`. If you want to use an image with the `Generalized` state, see [Create a VM from a generalized image version](#).

Create the VM using [az vm create](#) using the `--specialized` parameter to indicate that the image is a specialized image.

Use the `Id`, appended with `/Versions/latest` to use the latest version, as the value for `--image` to create a VM.

In this example, we're creating a VM from the latest version of the *myImageDefinition* image.

```
imgDef="/SharedGalleries/1a2b3c4d-1234-abcd-1a2b3c4d5e6f-  
MYDIRECTSHARED/Images/myDirectDefinition/Versions/latest"  
vmResourceGroup=myResourceGroup  
location=westus  
vmName=myVM  
  
az group create --name $vmResourceGroup --location $location  
  
az vm create\  
  --resource-group $vmResourceGroup \  
  --name $vmName \  
  --image $imgDef \  
  --specialized
```

## Next steps

- [Create an Azure Compute Gallery](#)
- [Create an image in an Azure Compute Gallery](#)

# List, update, and delete gallery resources

9/21/2022 • 6 minutes to read • [Edit Online](#)

You can manage your Azure Compute Gallery (formerly known as Shared Image Gallery) resources using the Azure CLI or Azure PowerShell.

## List galleries shared with you

- [CLI](#)
- [REST](#)
- [PowerShell](#)

List Galleries shared with your subscription.

```
region=westus
az sig list-shared --location $region
```

List Galleries shared with your tenant.

```
region=westus
az sig list-shared --location $region --shared-to tenant
```

The output will contain the public `name` and `uniqueID` of the gallery that is shared with you. You can use the name of the gallery to query for images that are available through the gallery.

Here is example output:

```
[  
 {  
   "location": "westus",  
   "name": "1231b567-8a99-1a2b-1a23-123456789abc-MYDIRECTSHARED",  
   "uniqueId": "/SharedGalleries/1231b567-8a99-1a2b-1a23-123456789abc-MYDIRECTSHARED"  
 }  
]
```

## Update resources

There are some limitations on what can be updated. The following items can be updated:

Azure Compute Gallery:

- Description

Image definition:

- Recommended vCPUs
- Recommended memory
- Description
- End of life date

Image version:

- Regional replica count
- Target regions
- Exclusion from latest
- End of life date

If you plan on adding replica regions, don't delete the source managed image. The source managed image is needed for replicating the image version to additional regions.

- [CLI](#)
- [PowerShell](#)

Update the description of a gallery using [az sig update](#).

```
az sig update \
--gallery-name myGallery \
--resource-group myGalleryRG \
--set description="My updated description."
```

Update the description of an image definition using [az sig image-definition update](#).

```
az sig image-definition update \
--gallery-name myGallery\
--resource-group myGalleryRG \
--gallery-image-definition myImageDefinition \
--set description="My updated description."
```

Update an image version to add a region to replicate to using [az sig image-version update](#). This change will take a while as the image gets replicated to the new region.

```
az sig image-version update \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--gallery-image-version 1.0.0 \
--add publishingProfile.targetRegions name=eastus
```

This example shows how to use [az sig image-version update](#) to exclude this image version from being used as the *latest* image.

```
az sig image-version update \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--gallery-image-version 1.0.0 \
--set publishingProfile.excludeFromLatest=true
```

This example shows how to use [az sig image-version update](#) to include this image version in being considered for *latest* image.

```
az sig image-version update \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--gallery-image-version 1.0.0 \
--set publishingProfile.excludeFromLatest=false
```

## Delete resources

You have to delete resources in reverse order, by deleting the image version first. After you delete all of the image versions, you can delete the image definition. After you delete all image definitions, you can delete the gallery.

- [CLI](#)
- [PowerShell](#)

Before you can delete a community shared gallery, you need to use [az sig share reset](#) to stop sharing the gallery publicly.

Delete an image version using [az sig image-version delete](#).

```
az sig image-version delete \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
--gallery-image-version 1.0.0
```

Delete an image definition using [az sig image-definition delete](#).

```
az sig image-definition delete \
--resource-group myGalleryRG \
--gallery-name myGallery \
--gallery-image-definition myImageDefinition
```

Delete a gallery using [az sig delete](#).

```
az sig delete \
--resource-group myGalleryRG \
--gallery-name myGallery
```

## Community galleries

### IMPORTANT

Azure Compute Gallery – community galleries is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery - community gallery](#).

To list your own galleries, and output the public names for your community galleries:

```
az sig list --query [*].{Name:name,PublicName:sharingProfile.communityGalleryInfo.publicNames}"
```

### NOTE

As an end user, to get the public name of a community gallery, you currently need to use the portal. Go to **Virtual machines > Create > Azure virtual machine > Image > See all images > Community Images > Public gallery name**.

List all of the image definitions that are available in a community gallery using [az sig image-definition list-community](#).

In this example, we list all of the images in the *ContosoImage* gallery in *West US* and by name, the unique ID that is needed to create a VM, OS and OS state.

```
az sig image-definition list-community \
--public-gallery-name "ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f" \
--location westus \
--query [*].{"Name:name, ID:uniqueId, OS:osType, State:osState}" -o table
```

List image versions shared in a community gallery using [az sig image-version list-community](#):

```
az sig image-version list-community \
--location westus \
--public-gallery-name "ContosoImages-1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f" \
--gallery-image-definition myImageDefinition \
--query [*].{"Name:name, UniqueId:uniqueId}" \
-o table
```

## Direct shared galleries

### IMPORTANT

Azure Compute Gallery – direct shared gallery is currently in PREVIEW and subject to the [Preview Terms for Azure Compute Gallery](#).

To publish images to a direct shared gallery during the preview, you need to register at <https://aka.ms/directsharedgallery-preview>. Creating VMs from a direct shared gallery is open to all Azure users.

During the preview, you need to create a new gallery, with the property `sharingProfile.permissions` set to `Groups`. When using the CLI to create a gallery, use the `--permissions groups` parameter. You can't use an existing gallery, the property can't currently be updated.

To find the `uniqueID` of a gallery that is shared with you, use [az sig list-shared](#). In this example, we are looking for galleries in the West US region.

```
region=westus
az sig list-shared --location $region --query "[].uniqueId" -o tsv
```

List all of the image definitions that are shared directly with you, use [az sig image-definition list-shared](#).

In this example, we list all of the images in the gallery in *West US* and by name, the unique ID that is needed to create a VM, OS and OS state.

```
name="1a2b3c4d-1234-abcd-1234-1a2b3c4d5e6f-myDirectShared"
az sig image-definition list-shared \
--gallery-unique-name $name
--location $region \
--query [*].{"Name:name, ID:uniqueId, OS:osType, State:osState}" -o table
```

List image versions directly shared to you using [az sig image-version list-shared](#):

```
imgDef="myImageDefinition"
az sig image-version list-shared \
--location $region \
--public-gallery-name $name \
--gallery-image-definition $imgDef \
--query "[*].\"{Name:name,UniqueId:uniqueId}\"" \
-o table
```

## Next steps

- Create an [image definition and an image version](#).
- Create a VM from a [generalized](#) or [specialized](#) image in a direct shared gallery.

# Troubleshoot images in an Azure Compute Gallery

9/21/2022 • 21 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

If you have problems performing any operations on Azure Compute Gallery (formerly known as Shared Image Gallery) resources, like galleries, image definitions, and image versions, run the failing command again in debug mode. You activate debug mode by passing the `--debug` switch with the Azure CLI and the `-Debug` switch with PowerShell. After you've located the error, follow this article to troubleshoot it.

## Creating or modifying a gallery

Error message	Cause	Mitigation
Gallery name is invalid. Allowed characters are English alphanumeric characters, with underscores, and periods allowed in the middle, up to 80 characters total. All other special characters, including dashes, are disallowed.	The name for the gallery does not meet the naming requirements	Choose a name that meets the following conditions: <ul style="list-style-type: none"><li>• Has an 80-character limit</li><li>• Contains only English letters, numbers, underscores, and periods</li><li>• Starts and ends with English letters or numbers</li></ul>
The provided resource name <galleryName> has these invalid trailing characters: <character>. The name can not end with characters: <character>	The name for the gallery ends with a period or underscore	Choose a name for the gallery that meets the following conditions: <ul style="list-style-type: none"><li>• Has an 80-character limit</li><li>• Contains only English letters, numbers, underscores, and periods</li><li>• Starts and ends with English letters or numbers</li></ul>
The provided location <region> is not available for resource type 'Microsoft.Compute/galleries'. List of available regions for the resource type is ...	The region specified for the gallery is incorrect or requires an access request	Check that the region name is correct. If the region name is correct, submit <a href="#">an access request</a> for the region
Can not delete resource before nested resources are deleted	You've tried to delete a gallery that contains at least one existing image definition. A gallery must be empty before it can be deleted	Delete all image definitions inside the gallery and then proceed to delete the gallery. If the image definition contains image versions, you must delete the image versions before you delete the image definitions
The gallery name <galleryName> is not unique within the subscription <subscriptionID>. Please pick another gallery name	You have an existing gallery with the same name and have tried to create another gallery with the same name	Choose a different name for the gallery
The resource <galleryName> already exists in location <region_1> in resource group <resourceGroup>. A resource with the same name cannot be created in location <region_2>. Please select a new resource name	You have an existing gallery with the same name and have tried to create another gallery with the same name	Choose a different name for the gallery

## Creating or modifying image definitions

Error message	Cause	Mitigation
Changing property 'galleryImage.properties.<property>' is not allowed	You tried to change the OS type, OS state, Hyper-V generation, offer, publisher, or SKU. Changing any of these properties is not permitted	Create a new image definition instead
The resource <galleryName/imageDefinitionName> already exists in location <region_1> in resource group <resourceGroup>. A resource with the same name cannot be created in location <region_2>. Please select a new resource name.	You have an existing image definition in the same gallery and resource group with the same name. You've tried to create another image definition with the same name and in the same gallery but in a different region.	Use a different name for the image definition, or put the image definition in a different gallery or resource group
The provided resource name <galleryName>/<imageDefinitionName> has these invalid trailing characters: <character>. The name can not end with characters: <character>	The <imageDefinitionName> name ends with a period or underscore.	Choose a name for the image definition that meets the following conditions: <ul style="list-style-type: none"> <li>Has an 80-character limit</li> <li>Contains only English letters, numbers, underscores, hyphens and periods</li> <li>Starts and ends with English letters or numbers</li> </ul>
The entity name <imageDefinitionName> is invalid according to its validation rule: ^[^\W][\w-_]{0,79}(?![.-])\$^*	The <imageDefinitionName> name ends with a period or underscore.	Choose a name for the image definition that meets the following conditions: <ul style="list-style-type: none"> <li>Has an 80-character limit</li> <li>Contains only English letters, numbers, underscores, hyphens, and periods</li> <li>Starts and ends with English letters or numbers</li> </ul>
Asset name galleryImage.properties.identifier. <property> is not valid. It cannot be empty. Allowed characters are uppercase or lowercase letters, digits, hyphen(-), period (), underscore (_). Names are not allowed to end with period(.). The length of the name cannot exceed <number> characters.	The publisher, offer, or SKU value does not meet the naming requirements.	Choose a value that meets the following conditions: <ul style="list-style-type: none"> <li>Has a 128-character limit for publisher or 64-character limit for offer and SKU</li> <li>Contains only English letters, numbers, hyphens, underscores, and periods</li> <li>Does not end with a period</li> </ul>
Can not perform requested operation on nested resource. Parent resource <galleryName> not found.	There is no gallery with the name <galleryName> in the current subscription and resource group.	Check that the names of the gallery, subscription, and resource group are correct. Otherwise, create a new gallery named <galleryName>
The provided location <region> is not available for resource type 'Microsoft.Compute/galleries'. List of available regions for the resource type is ...	The <region> name is incorrect or requires an access request.	Check that the region name is spelled correctly. You can run this command to see what regions you have access to. If the region is not in the list, submit <a href="#">an access request</a> .

ERROR MESSAGE	CAUSE	MITIGATION
Unable to serialize value: <value> as type: 'iso-8601', ISO8601Error: ISO 8601 time designator 'T' missing. Unable to parse datetime string <value>	The value provided to the property is not properly formatted as a date.	Provide a date in the yyyy-MM-dd, yyyy-MM-dd'T'HH:mm:sszzz, or <a href="#">ISO 8601</a> -valid format.
Could not convert string to DateTimeOffset: <value>. Path 'properties.<property>'	The value provided to the property is not properly formatted as a date.	Provide a date in the yyyy-MM-dd, yyyy-MM-dd'T'HH:mm:sszzz, or <a href="#">ISO 8601</a> -valid format
EndOfLifeDate must be set to a future date.	The end-of-life date property is not properly formatted as a date that's after today's date.	Provide a date in the yyyy-MM-dd, yyyy-MM-dd'T'HH:mm:sszzz, or <a href="#">ISO 8601</a> -valid format
<i>argument --&lt;property&gt;: invalid int value: &lt;value&gt;</i>	<property> accepts only integer values, and <value> is not an integer.	Choose an integer value.
The minimum value of <property> must not be greater than the maximum value of <property>.	The minimum value provided for <property> is higher than the maximum value provided for <property>.	Change the values so that the minimum is less than or equal to the maximum.
Gallery image: <imageDefinitionName> identified by (publisher:<Publisher>, offer:<Offer>, sku:<SKU>) already exists. Choose a different publisher, offer, sku combination.	You've tried to create a new image definition with the same publisher, offer, and SKU triplet as an existing image definition in the same gallery.	Within a gallery, all image definitions must have a unique combination of publisher, offer, and SKU. Choose a unique combination, or choose a new gallery and create the image definition again
Can not delete resource before nested resources are deleted.	You've tried to delete an image definition that contains image versions. An image definition must be empty before it can be deleted.	Delete all image versions inside the image definition and then proceed to delete the image definition
Cannot bind parameter <property>. Cannot convert value <value> to type <propertyType>. Unable to match the identifier name <value> to a valid enumerator name. Specify one of the following enumerator names and try again: <choice_1>, <choice_2>, ...	The property has a restricted list of possible values, and <value> is not one of them	Choose one of the possible <choice> values
Cannot bind parameter <property>. Cannot convert value <value> to type "System.DateTime"	The value provided to the property is not properly formatted as a date.	Provide a date in the yyyy-MM-dd, yyyy-MM-dd'T'HH:mm:sszzz, or <a href="#">ISO 8601</a> -valid format
Cannot bind parameter <property>. Cannot convert value <value> to type "System.Int32"	<property> accepts only integer values, and <value> is not an integer.	Choose an integer value
ZRS storage account type is not supported in this region.	You've chosen standard zone-redundant storage (ZRS) in a region that does not yet support it	Change the storage account type to <a href="#">Premium_LRS</a> or <a href="#">Standard_LRS</a> . Check our documentation for the latest <a href="#">list of regions</a> with ZRS preview enabled

## Creating or updating image versions

Error message	Cause	Mitigation
The provided location <region> is not available for resource type 'Microsoft.Compute/galleries'. List of available regions for the resource type is	The <region> name is incorrect or requires an access request	Check that the region name is spelled correctly. You can run this command to see what regions you have access to. If the region is not in the list, submit <a href="#">an access request</a>
Can not perform requested operation on nested resource. Parent resource <galleryName/imageDefinitionName> not found	There is no gallery with the name <galleryName/imageDefinitionName> in the current subscription and resource group	Check that the names of the gallery, subscription, and resource group are correct. Otherwise, create a new gallery with the name <galleryName> and/or an image definition named <imageDefinitionName> in the indicated resource group
Cannot bind parameter <property>. Cannot convert value <value> to type "System.DateTime"	The value provided to the property is not properly formatted as a date	Provide a date in the yyyy-MM-dd, yyyy-MM-dd'T'HH:mm:sszzz, or <a href="#">ISO 8601</a> -valid format
Cannot bind parameter <property>. Cannot convert value <value> to type "System.Int32"	<property> accepts only integer values, and <value> is not an integer	Choose an integer value
Gallery image version publishing profile regions <publishingRegions> must contain the location of image version <sourceRegion>	The location of the source image (<sourceRegion>) must be included in the <publishingRegions> list	Include <sourceRegion> in the <publishingRegions> list
The value <value> of parameter <property> is out of range. The value must be between <minValue> and <maxValue>, inclusive	<value> is outside the range of possible values for <property>	Choose a value that's within the range of <minValue> and <maxValue>, inclusive
Source <resourceID> is not found. Please check source exists, and is in the same region as gallery image version being created	There is no source located at <resourceID>, or the source at <resourceID> is not in the same region as the gallery image being created	Check that the <resourceID> value is correct and that the source region of the gallery image version is the same as the region of the <resourceID> value
Changing property 'galleryImageVersion.properties.storageProfile.<diskImage>.source.id' is not allowed	The source ID of a gallery image version can't be changed after creation	Ensure that the source ID is the same as the existing source ID, change the version number of the image version, or delete the current image version and try again
Duplicated lun numbers have been detected in the input data disks. Lun number must be unique for each data disk	When you're creating an image version by using a list of disks and/or disk snapshots, two or more disks or disk snapshots have the same LUN	Remove or change any duplicate LUNs
Duplicated source ids are found in the input disks. Source id should be unique for each disk	When you're creating an image version by using a list of disks and/or disk snapshots, two or more disks or disk snapshots have the same resource ID	Remove or change any duplicate disk source IDs

ERROR MESSAGE	CAUSE	MITIGATION
Property id <resourceID> at path 'properties.storageProfile.<diskImages>.source.id' is invalid. Expect fully qualified resource Id that start with '/subscriptions/<subscriptionID>' or '/providers/<resourceProviderNamespace>/'	The <resourceID> value is incorrectly formatted	Check that the resource ID is correct
The source id: <resourceID> must either be a managed image, virtual machine or another gallery image version	The <resourceID> value is incorrectly formatted	If you're using a VM, managed image, or gallery image version as the source image, check that the resource ID of the VM, managed image, or gallery image version is correct
The source id: <resourceID> must be a managed disk or snapshot	The <resourceID> value is incorrectly formatted	If you're using disks and/or disk snapshots as sources for the image version, check that the resource IDs of the disks and/or disk snapshots are correct
Cannot create Gallery Image Version from: <resourceID> since the OS State in the parent gallery image (<OsState_1>) is not <OsState_2>	The operating system state (Generalized or Specialized) does not match the operating system state specified in the image definition	Either choose a source based on a VM with the operating system state of <OsState_1> or create a new image definition for VMs based on <OsState_2>
The resource with id '<resourceID>' has a different Hypervisor generation ['<V#_1>'] than the parent gallery image Hypervisor generation ['<V#_2>']	The hypervisor generation of the image version does not match the hypervisor generation specified in the image definition. The image definition operating system is <V#_1>, and the image version operating system is <V#_2>	Either choose a source with the same hypervisor generation as the image definition or create/choose a new image definition that has the same hypervisor generation as the image version
The resource with id '<resourceID>' has a different OS type ['<OsType_1>'] than the parent gallery image OS type generation ['<OsType_2>']	The hypervisor generation of the image version does not match the hypervisor generation specified in the image definition. The image definition operating system is <OsType_1>, and the image version operating system is <OsType_2>	Either choose a source with the same operating system (Linux/Windows) as the image definition or create/choose a new image definition that has the same operating system generation as the image version
Source virtual machine <resourceID> cannot contain an ephemeral OS disk	The source at <resourceID> contains an ephemeral OS disk. The Azure Compute Gallery does not currently support ephemeral OS disks	Choose a different source based on a VM that does not use an ephemeral OS disk
Source virtual machine <resourceID> cannot contain disk ['<diskID>'] stored in an UltraSSD account type	The disk <diskID> is an Ultra SSD disk. The Azure Compute Gallery does not currently support Ultra SSD disks	Use a source that contains only Premium SSD, Standard SSD, and/or Standard HDD managed disks
Source virtual machine <resourceID> must be created from Managed Disks	The virtual machine in <resourceID> uses unmanaged disks	Use a source based on a VM that contains only Premium SSD, Standard SSD, and/or Standard HDD managed disks

ERROR MESSAGE	CAUSE	MITIGATION
Too many requests on source '<resourceID>'. Please reduce the number of requests on the source or wait some time before retrying	The source for this image version is currently being throttled because of too many requests	Try to create the image version later
The disk encryption set '<diskEncryptionSetID>' must be in the same subscription '<subscriptionID>' as the gallery resource	Disk encryption sets can be used only in the same subscription and region in which they were created	Create or use an encryption set in the same subscription and region as the image version
Encrypted source: '<resourceID>' is in a different subscription ID than the current gallery image version subscription '<subscriptionID_1>'. Please retry with an unencrypted source(s) or use the source's subscription '<subscriptionID_2>' to create the gallery image version	The Azure Compute Gallery does not currently support creating image versions in another subscription from another source image if the source image is encrypted	Use an unencrypted source or create the image version in the same subscription as the source
The disk encryption set <diskEncryptionSetID> was not found	The disk encryption might be incorrect	Check that the resource ID of the disk encryption set is correct
The image version name is invalid. The image version name should follow Major(int).Minor(int).Patch(int) format, for e.g: 1.0.0, 2018.12.1 etc	The valid format for an image version is three integers separated by a period. The image version name did not meet the valid format	Use an image version name that follows the format Major(int).Minor(int).Patch(int). For example: 1.0.0. or 2018.12.1
The value of parameter galleryArtifactVersion.properties.publishingProfile.targetRegions.encryption.dataDiskImages.diskEncryptionSetId is invalid	The resource ID of the disk encryption set used on a data disk image uses an invalid format	Ensure that the resource ID of the disk encryption set follows the format /subscriptions/<subscriptionID>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/<diskEncryptionSetName>
The value of parameter galleryArtifactVersion.properties.publishingProfile.targetRegions.encryption.osDiskImage.diskEncryptionSetId is invalid	The resource ID of the disk encryption set used on the OS disk image uses an invalid format	Ensure that the resource ID of the disk encryption set follows the format /subscriptions/<subscriptionID>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/<diskEncryptionSetName>
Cannot specify new data disk image encryption lun [<number>] with a disk encryption set in region [<region>] for update gallery image version request. To update this version, remove the new lun. If you need to change the data disk image encryption settings, you must create a new gallery image version with the correct settings.	You added encryption to the data disk of an existing image version. You can't add encryption to an existing image version	Create a new gallery image version or remove the added encryption settings
The gallery artifact version source can only be specified either directly under storageProfile or within individual OS or data disks. One and only one source type (user image, snapshot, disk, virtual machine) can be provided	The source ID is missing	Ensure that the source ID of the source is present

ERROR MESSAGE	CAUSE	MITIGATION
Source was not found: <resourceID>. Please make sure the source exists	The resource ID of the source might be incorrect	Ensure that the resource ID of the source is correct
A disk encryption set is required for disk 'galleryArtifactVersion.properties.publishingProfile.targetRegions.encryption.osDiskImage.diskEncryptionSetId' in target region '<region_1>' since disk encryption set '<diskEncryptionSetID>' is used for the corresponding disk in region '<region_2>'	Encryption has been used on the OS disk in <region_2>, but not in <region_1>	If you're using encryption on the OS disk, use encryption in all regions
A disk encryption set is required for disk 'LUN <number>' in target region '<region_1>' since disk encryption set '<diskEncryptionSetID>' is used for the corresponding disk in region '<region_2>'	Encryption has been used on the data disk at LUN <number> in <region_2>, but not in <region_1>	If you're using encryption on a data disk, use encryption in all regions
An invalid lun [<number>] was specified in encryption.dataDiskImages. The lun must be one of the following values ['0,9']	The LUN specified for the encryption does not match any of the LUNs for disks attached to the VM	Change the LUN in the encryption to the LUN of a data disk present in the VM
Duplicate luns '<number>' were specified in target region '<region>' encryption.dataDiskImages	The encryption settings used in <region> specified a LUN at least twice	Change the LUN in <region> to make sure that all the LUNs are unique in <region>
OSDiskImage and DataDiskImage cannot point to same blob <sourceID>	The sources for the OS disk and at least one data disk are not unique	Change the source for the OS disk and/or data disks to ensure that the OS disk as well as each data disk is unique
Duplicate regions are not allowed in target publishing regions	A region is listed among the publishing regions more than once	Remove the duplicate region
Adding new Data Disks or changing the LUN of a Data Disk in an existing Image is not allowed	An update call to the image version either contains a new data disk or has a new LUN for a disk	Use the LUNs and data disks of the existing image version
The disk encryption set <diskEncryptionSetID> must be in the same subscription <subscriptionID> as the gallery resource	The Azure Compute Gallery does not currently support using a disk encryption set in a different subscription	Create the image version and disk encryption set in the same subscription
Replication failed in this region due to 'The GalleryImageVersion source resource size 2048 exceeds the max size 1024 supported	A data disk in the source is greater than 1TB	Resize the data disk to under 1 TB

ERROR MESSAGE	CAUSE	MITIGATION
Operation 'Update Gallery Image Version' is not allowed on <versionNumber>; since it is marked for deletion. You can only retry the Delete operation (or wait for an ongoing one to complete)	You attempted to update a gallery image version that is in the process of being deleted	Wait for the deletion event to complete and recreate the image version again
Encryption is not supported for source resource '<sourceID>'. Please use a different source resource type which supports encryption or remove encryption properties	Currently the Azure Compute Gallery only supports encryption for VMs, disks, snapshots and managed images. One of the sources provided for the image version is not in the previous list of sources that support encryption	Remove the disk encryption set from the image version and contact the support team.

## Creating or updating a VM or scale sets from an image version

ERROR MESSAGE	CAUSE	MITIGATION
There is no latest image version exists for "<imageDefinitionResourceId>"	The image definition you used to deploy the virtual machine does not contain any image versions that are included in latest	Ensure that there is at least one image version that has 'Exclude from latest' set to False
The gallery image /subscriptions/<subscriptionID>/resourceGroups/<resourceGroup>/providers/Microsoft.Compute/galleries/<galleryName>/images/<imageName>/versions/<versionNumber> is not available in <region> region. Please contact image owner to replicate to this region, or change your requested region.	The version selected for deployment does not exist or does not have a replica in the indicated region	Ensure that the name of the image resource is correct and that there is at least one replica in the indicated region
The gallery image /subscriptions/<subscriptionID>/resourceGroups/<resourceGroup>/providers/Microsoft.Compute/galleries/<galleryName>/images/<imageName> is not available in <region> region. Please contact image owner to replicate to this region, or change your requested region	The image definition selected for deployment does not have any image versions that are included in latest and also in the indicated region	Ensure that there is at least one image version in the region that has 'Exclude from latest' set to False
The client has permission to perform action 'Microsoft.Compute/galleries/images/versions/read' on scope <resourceID>, however the current tenant <tenantID> is not authorized to access linked subscription <subscriptionID>	The virtual machine or scale set was created through a gallery image in another tenant. You've tried to make a change to the virtual machine or scale set, but you don't have access to the subscription that owns the image	Contact the owner of the subscription of the image version to grant read access to the image version

ERROR MESSAGE	CAUSE	MITIGATION
The gallery image <resourceID> is not available in <region> region. Please contact image owner to replicate to this region, or change your requested region.	The VM is being created in a region that's not among the list of published regions for the gallery image.	Either replicate the image to the region or create a VM in one of the regions in the gallery image's publishing regions
Parameter 'osProfile' is not allowed	Admin username, password, or SSH keys were provided for a VM that was created from a specialized image version	Don't include the admin username, password, or SSH keys if you intend to create a VM from that image. Otherwise, use a generalized image version and supply the admin username, password, or SSH keys
Required parameter 'osProfile' is missing (null)	The VM is created from a generalized image, and it's missing the admin username, password, or SSH keys. Because generalized images don't retain the admin username, password, or SSH keys, these fields must be specified during creation of a VM or scale set	Specify the admin username, password, or SSH keys, or use a specialized image version
Cannot update Virtual Machine Scale Set <vmssName> as the current OS state of the VM Scale Set is Generalized which is different from the updated gallery image OS state which is Specialized	The current source image for the scale set is a generalized source image, but it's being updated with a source image that is specialized. The current source image and the new source image for a scale set must be of the same state	To update the scale set, use a generalized image version
Disk encryption set <diskEncryptionSetID> in Azure Compute Gallery <versionID> belongs to subscription <subscriptionID_1> and cannot be used with resource " in subscription <subscriptionID_2>	The disk encryption set used to encrypt the image version resides in a different subscription than the subscription to host the image version	Use the same subscription for the image version and disk encryption set
The resource with id <vmID> has a different plan [{"name": " <name>","publisher": " <publisher>","product": " <product>","promotionCode": " <promotionCode>"}] than the parent gallery image plan [null]	The parent image definition for the image version being deployed does not have a purchase plan information	Create an image definition with the same purchase plan details from the error message and create the image version within the image definition
The VM or virtual machine scale set creation takes a long time	NA	Verify that the <b>OSType</b> of the image version that you're trying to create the VM or virtual machine scale set from has the same <b>OSType</b> of the source that you used to create the image version

## Creating a disk from an image version

ERROR MESSAGE	CAUSE	MITIGATION
The value of parameter imageReference is invalid.	You've tried to export from a gallery Image version to a disk but used a LUN position that does not exist on the image.	Check the image version to see what LUN positions are in use

## Sharing resources

The sharing of gallery, image definition, and image version resources across subscriptions is enabled using [Azure role-based access control \(Azure RBAC\)](#).

## Replication speed

Use the `--expand ReplicationStatus` flag to check if the replication to all the specified target regions has finished. If not, wait for up to six hours for the job to finish. If it fails, trigger the command again to create and replicate the image version. If there are many target regions that the image version is being replicated to, consider doing the replication in phases.

## Azure limits and quotas

[Azure limits and quotas](#) apply to all Azure Compute Gallery, image definition, and image version resources. Make sure you're within the limits for your subscriptions.

## Next steps

Learn more about [Azure Compute Galleries](#).

# Use customer-managed keys for encrypting images

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Images in an Azure Compute Gallery (formerly known as Shared Image Gallery) are stored as snapshots, so they're automatically encrypted through server-side encryption. Server-side encryption uses 256-bit [AES encryption](#), one of the strongest block ciphers available. Server-side encryption is also FIPS 140-2 compliant. For more information about the cryptographic modules underlying Azure managed disks, see [Cryptography API: Next Generation](#).

You can rely on platform-managed keys for the encryption of your images, or use your own keys. You can also use both together, for double encryption. If you choose to manage encryption with your own keys, you can specify a *customer-managed key* to use for encrypting and decrypting all disks in your images.

Server-side encryption through customer-managed keys uses Azure Key Vault. You can either import [your RSA keys](#) to your key vault or generate new RSA keys in Azure Key Vault.

## Prerequisites

This article requires you to already have a disk encryption set in each region where you want to replicate your image:

- To use only a customer-managed key, see the articles about enabling customer-managed keys with server-side encryption by using the [Azure portal](#) or [PowerShell](#).
- To use both platform-managed and customer-managed keys (for double encryption), see the articles about enabling double encryption at rest by using the [Azure portal](#) or [PowerShell](#).

### IMPORTANT

You must use the link <https://aka.ms/diskencryptionupdates> to access the Azure portal. Double encryption at rest is not currently visible in the public Azure portal unless you use that link.

## Limitations

When you're using customer-managed keys for encrypting images in an Azure Compute Gallery, these limitations apply:

- Encryption key sets must be in the same subscription as your image.
- Encryption key sets are regional resources, so each region requires a different encryption key set.
- You can't copy or share images that use customer-managed keys.
- After you've used your own keys to encrypt a disk or image, you can't go back to using platform-managed keys for encrypting those disks or images.

## PowerShell

To specify a disk encryption set for an image version, use `New-AzGalleryImageVersion` with the `-TargetRegion` parameter:

```



```

## Create a VM

You can create a virtual machine (VM) from an Azure Compute Gallery and use customer-managed keys to encrypt the disks. The syntax is the same as creating a [generalized](#) or [specialized](#) VM from an image. Use the extended parameter set and add

```
Set-AzVMOSDisk -Name $($vmName +"_OSDisk") -DiskEncryptionSetId $diskEncryptionSet.Id -CreateOption FromImage
```

to the VM configuration.

For data disks, add the `-DiskEncryptionSetId $setID` parameter when you use [Add-AzVMDataDisk](#).

## CLI

To specify a disk encryption set for an image version, use [az image gallery create-image-version](#) with the `--target-region-encryption` parameter. The format for `--target-region-encryption` is a comma-separated list of keys for encrypting the OS and data disks. It should look like this:

```
<encryption set for the OS disk>,<Lun number of the data disk>,<encryption set for the data disk>,<Lun number for the second data disk>,<encryption set for the second data disk>
```

If the source for the OS disk is a managed disk or a VM, use `--managed-image` to specify the source for the image version. In this example, the source is a managed image that has an OS disk and a data disk at LUN 0. The OS disk will be encrypted with DiskEncryptionSet1, and the data disk will be encrypted with DiskEncryptionSet2.

```
az sig image-version create \
-g MyResourceGroup \
--gallery-image-version 1.0.0 \
--location westus \
--target-regions westus=2=standard_lrs eastus2 \
--target-region-encryption WestUSDiskEncryptionSet1,0,WestUSDiskEncryptionSet2
EastUS2DiskEncryptionSet1,0,EastUS2DiskEncryptionSet2 \
--gallery-name MyGallery \
--gallery-image-definition MyImage \
--managed-image "/subscriptions/<subscription
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/images/myImage"
```

If the source for the OS disk is a snapshot, use `--os-snapshot` to specify the OS disk. If there are data disk snapshots that should also be part of the image version, add those. Use `--data-snapshot-luns` to specify the LUN, and use `--data-snapshots` to specify the snapshots.

In this example, the sources are disk snapshots. There's an OS disk and a data disk at LUN 0. The OS disk will be encrypted with DiskEncryptionSet1, and the data disk will be encrypted with DiskEncryptionSet2.

```
az sig image-version create \
-g MyResourceGroup \
--gallery-image-version 1.0.0 \
--location westus \
--target-regions westus=2=standard_lrs eastus \
--target-region-encryption WestUSDiskEncryptionSet1,0,WestUSDiskEncryptionSet2
EastUS2DiskEncryptionSet1,0,EastUS2DiskEncryptionSet2 \
--os-snapshot "/subscriptions/<subscription
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/snapshots/myOSSnapshot" \
--data-snapshot-luns 0 \
--data-snapshots "/subscriptions/<subscription
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/snapshots/myDDSnapshot" \
--gallery-name MyGallery \
--gallery-image-definition MyImage
```

## Create the VM

You can create a VM from an Azure Compute Gallery and use customer-managed keys to encrypt the disks. The syntax is the same as creating a [generalized](#) or [specialized](#) VM from an image. Just add the `--os-disk-encryption-set` parameter with the ID of the encryption set. For data disks, add `--data-disk-encryption-sets` with a space-delimited list of the disk encryption sets for the data disks.

## Portal

When you create your image version in the portal, you can use the **Encryption** tab to apply your storage encryption sets.

1. On the [Create an image version](#) page, select the **Encryption** tab.
2. In **Encryption type**, select **Encryption at-rest with a customer-managed key** or **Double encryption with platform-managed and customer-managed keys**.
3. For each disk in the image, select an encryption set from the **Disk encryption set** drop-down list.

## Create the VM

You can create a VM from an image version and use customer-managed keys to encrypt the disks. When you create the VM in the portal, on the **Disks** tab, select **Encryption at-rest with customer-managed keys** or **Double encryption with platform-managed and customer-managed keys** for **Encryption type**. You can then select the encryption set from the drop-down list.

## Next steps

Learn more about [server-side disk encryption](#).

For information about how to supply purchase plan information, see [Supply Azure Marketplace purchase plan information when creating images](#).

# Export an image version to a managed disk

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can export the OS or a single data disk from an image version as a managed disk from an image version stored in an Azure Compute Gallery (formerly known as Shared Image Gallery).

## CLI

List the image versions in a gallery using [az sig image-version list](#). In this example, we are looking for all of the image versions that are part of the *myImageDefinition* image definition in the *myGallery* gallery.

```
az sig image-version list \
--resource-group myResourceGroup\
--gallery-name myGallery \
--gallery-image-definition myImageDefinition \
-o table
```

Set the `source` variable to the ID of the image version, then use [az disk create](#) to create the managed disk.

In this example, we export the OS disk of the image version to create a managed disk named *myManagedOSDisk*, in the *EastUS* region, in a resource group named *myResourceGroup*.

```
source="/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/galleries/<galleryName>/images/<galleryImageDefinition>/versions/<imageVersion>"  
  
az disk create --resource-group myResourceGroup --location EastUS --name myManagedOSDisk --gallery-image-reference $source
```

If you want to export a data disk from the image version, add `--gallery-image-reference-lun` to specify the LUN location of the data disk to export.

In this example, we export the data disk located at LUN 0 of the image version to create a managed disk named *myManagedDataDisk*, in the *EastUS* region, in a resource group named *myResourceGroup*.

```
source="/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/galleries/<galleryName>/images/<galleryImageDefinition>/versions/<imageVersion>"  
  
az disk create --resource-group myResourceGroup --location EastUS --name myManagedDataDisk --gallery-image-reference $source --gallery-image-reference-lun 0
```

## PowerShell

List the image versions in a gallery using [Get-AzResource](#).

```
Get-AzResource `
-ResourceType Microsoft.Compute/galleries/images/versions | `
Format-Table -Property Name,ResourceId,ResourceGroupName
```

Once you have all of the information you need, you can use [Get-AzGalleryImageVersion](#) to get the source image

version you want to use and assign it to a variable. In this example, we are getting the `1.0.0` image version, of the `myImageDefinition` definition, in the `myGallery` source gallery, in the `myResourceGroup` resource group.

```
$sourceImgVer = Get-AzGalleryImageVersion `  
    -GalleryImageDefinitionName myImageDefinition `  
    -GalleryName myGallery `  
    -ResourceGroupName myResourceGroup `  
    -Name 1.0.0
```

After setting the `source` variable to the ID of the image version, use [New-AzDiskConfig](#) to create a disk configuration and [New-AzDisk](#) to create the disk.

In this example, we export the OS disk of the image version to create a managed disk named `myManagedOSDisk`, in the `EastUS` region, in a resource group named `myResourceGroup`.

Create a disk configuration.

```
$diskConfig = New-AzDiskConfig `  
    -Location EastUS `  
    -CreateOption FromImage `  
    -GalleryImageReference @{Id = $sourceImgVer.Id}
```

Create the disk.

```
New-AzDisk -Disk $diskConfig `  
    -ResourceGroupName myResourceGroup `  
    -DiskName myManagedOSDisk
```

If you want to export a data disk on the image version, add a LUN ID to the disk configuration to specify the LUN location of the data disk to export.

In this example, we export the data disk located at LUN 0 of the image version to create a managed disk named `myManagedDataDisk`, in the `EastUS` region, in a resource group named `myResourceGroup`.

Create a disk configuration.

```
$diskConfig = New-AzDiskConfig `  
    -Location EastUS `  
    -CreateOption FromImage `  
    -GalleryImageReference @{Id = $sourceImgVer.Id; Lun=0}
```

Create the disk.

```
New-AzDisk -Disk $diskConfig `  
    -ResourceGroupName myResourceGroup `  
    -DiskName myManagedDataDisk
```

## Next steps

You can also create an [image version](#) from a managed disk.

# Supply Azure Marketplace purchase plan information when creating images

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

If you are creating an image in a shared gallery, using a source that was originally created from an Azure Marketplace image, you may need to keep track of purchase plan information. This article shows how to find purchase plan information for a VM, then use that information when creating an image definition. We also cover using the information from the image definition to simplify supplying the purchase plan information when creating a VM for an image.

For more information about finding and using Marketplace images, see [Find and use Azure Marketplace images](#).

## Get the source VM information

If you still have the original VM, you can get the plan name, publisher, and product information from it using `Get-AzVM`. This example gets a VM named `myVM` in the `myResourceGroup` resource group and then displays the purchase plan information for the VM.

```
$vm = Get-AzVM `  
-ResourceGroupName myResourceGroup `  
-Name myVM  
$vm.Plan
```

## Create the image definition

Get the gallery you want to use to store the image. You can list all of the galleries first.

```
Get-AzResource -ResourceType Microsoft.Compute/galleries | Format-Table
```

Then create variables for the gallery you want to use. In this example, we are creating a variable named `$gallery` for `myGallery` in the `myGalleryRG` resource group.

```
$gallery = Get-AzGallery `  
-Name myGallery `  
-ResourceGroupName myGalleryRG
```

Create the image definition, using the `-PurchasePlanPublisher`, `-PurchasePlanProduct`, and `-PurchasePlanName` parameters.

```
$imageDefinition = New-AzGalleryImageDefinition `  
    -GalleryName $gallery.Name `  
    -ResourceGroupName $gallery.ResourceGroupName `  
    -Location $gallery.Location `  
    -Name 'myImageDefinition' `  
    -OsState specialized `  
    -OsType Linux `  
    -Publisher 'myPublisher' `  
    -Offer 'myOffer' `  
    -Sku 'mySKU' `  
    -PurchasePlanPublisher $vm.Plan.Publisher `  
    -PurchasePlanProduct $vm.Plan.Product `  
    -PurchasePlanName $vm.Plan.Name
```

Then create your [image version](#) using [New-AzGalleryImageVersion](#).

## Create the VM

When you go to create a VM from the image, you can use the information from the image definition to pass in the publisher information using [Set-AzVMPlan](#).

```

# Create some variables for the new VM.
$resourceGroup = "mySIGPubVM"
$location = "West Central US"
$vmName = "mySIGPubVM"

# Create a resource group
New-AzResourceGroup -Name $resourceGroup -Location $location

# Create the network resources.
$subnetConfig = New-AzVirtualNetworkSubnetConfig `

    -Name mySubnet `

    -AddressPrefix 192.168.1.0/24
$vnet = New-AzVirtualNetwork `

    -ResourceGroupName $resourceGroup `

    -Location $location `

    -Name MYvNET `

    -AddressPrefix 192.168.0.0/16 `

    -Subnet $subnetConfig
$pip = New-AzPublicIpAddress `

    -ResourceGroupName $resourceGroup `

    -Location $location `

    -Name "mynpublicdns$(Get-Random)" `

    -AllocationMethod Static `

    -IdleTimeoutInMinutes 4
$nsgRuleRDP = New-AzNetworkSecurityRuleConfig `

    -Name myNetworkSecurityGroupRuleRDP `

    -Protocol Tcp `

    -Direction Inbound `

    -Priority 1000 `

    -SourceAddressPrefix * `

    -SourcePortRange * `

    -DestinationAddressPrefix * `

    -DestinationPortRange 3389 -Access Deny
$nsg = New-AzNetworkSecurityGroup `

    -ResourceGroupName $resourceGroup `

    -Location $location `

    -Name myNetworkSecurityGroup `

    -SecurityRules $nsgRuleRDP
$nic = New-AzNetworkInterface `

    -Name $vmName `

    -ResourceGroupName $resourceGroup `

    -Location $location `

    -SubnetId $vnet.Subnets[0].Id `

    -PublicIpAddressId $pip.Id `

    -NetworkSecurityGroupId $nsg.Id

# Create a virtual machine configuration using Set-AzVMSourceImage -Id $imageDefinition.Id to use the latest
available image version. Set-AZVMPlan is used to pass the plan information in for the VM.

$vmConfig = New-AzVMConfig `

    -VMName $vmName `

    -VMSize Standard_D1_v2 | `

Set-AzVMSourceImage -Id $imageDefinition.Id | `

Set-AzVMPlan `

    -Publisher $imageDefinition.PurchasePlan.Publisher `

    -Product $imageDefinition.PurchasePlan.Product `

    -Name $imageDefinition.PurchasePlan.Name | `

Add-AzVMNetworkInterface -Id $nic.Id

# Create the virtual machine
New-AzVM `

    -ResourceGroupName $resourceGroup `

    -Location $location `

    -VM $vmConfig

```

## Next steps

For more information about finding and using Marketplace images, see [Find and use Azure Marketplace images](#).

# VM Applications overview

9/21/2022 • 14 minutes to read • [Edit Online](#)

VM Applications are a resource type in Azure Compute Gallery (formerly known as Shared Image Gallery) that simplifies management, sharing, and global distribution of applications for your virtual machines.

## IMPORTANT

Deploying VM applications in Azure Compute Gallery **do not currently support using Azure policies**.

While you can create an image of a VM with apps pre-installed, you would need to update your image each time you have application changes. Separating your application installation from your VM images means there's no need to publish a new image for every line of code change.

Application packages provide benefits over other deployment and packaging methods:

- Grouping and versioning of your packages
- VM applications can be globally replicated to be closer to your infrastructure, so you don't need to use AzCopy or other storage copy mechanisms to copy the bits across Azure regions.
- Sharing with other users through Azure Role Based Access Control (RBAC)
- Support for virtual machines, and both flexible and uniform scale sets
- If you have Network Security Group (NSG) rules applied on your VM or scale set, downloading the packages from an internet repository might not be possible. And with storage accounts, downloading packages onto locked-down VMs would require setting up private links.

## What are VM app packages?

The VM application packages use multiple resource types:

RESOURCE	DESCRIPTION
Azure compute gallery	A gallery is a repository for managing and sharing application packages. Users can share the gallery resource and all the child resources will be shared automatically. The gallery name must be unique per subscription. For example, you may have one gallery to store all your OS images and another gallery to store all your VM applications.
VM application	The definition of your VM application. It's a <i>logical</i> resource that stores the common metadata for all the versions under it. For example, you may have an application definition for Apache Tomcat and have multiple versions within it.
VM Application version	The deployable resource. You can globally replicate your VM application versions to target regions closer to your VM infrastructure. The VM Application Version must be replicated to a region before it may be deployed on a VM in that region.

## Limitations

- **No more than 3 replicas per region:** When creating a VM Application version, the maximum number of replicas per region is three.
- **Retrying failed installations:** Currently, the only way to retry a failed installation is to remove the application from the profile, then add it back.
- **Only 5 applications per VM:** No more than five applications may be deployed to a VM at any point.
- **1GB application size:** The maximum file size of an application version is 1 GB.
- **No guarantees on reboots in your script:** If your script requires a reboot, the recommendation is to place that application last during deployment. While the code attempts to handle reboots, it may fail.
- **Requires a VM Agent:** The VM agent must exist on the VM and be able to receive goal states.
- **Multiple versions of same application on the same VM:** You can't have multiple versions of the same application on a VM.
- **Move operations currently not supported:** Moving VMs with VM Apps to other resource groups are not supported at this time.

## Cost

There's no extra charge for using VM Application Packages, but you'll be charged for the following resources:

- Storage costs of storing each package and any replicas.
- Network egress charges for replication of the first image version from the source region to the replicated regions. Subsequent replicas are handled within the region, so there are no extra charges.

For more information on network egress, see [Bandwidth pricing](#).

## VM applications

The VM application resource defines the following about your VM application:

- Azure Compute Gallery where the VM application is stored
- Name of the application
- Supported OS type like Linux or Windows
- A description of the VM application

## VM application versions

VM application versions are the deployable resource. Versions are defined with the following properties:

- Version number
- Link to the application package file in a storage account
- Install string for installing the application
- Remove string to show how to properly remove the app
- Package file name to use when it's downloaded to the VM
- Configuration file name to be used to configure the app on the VM
- A link to the configuration file for the VM application, which you can include license files
- Update string for how to update the VM application to a newer version
- End-of-life date. End-of-life dates are informational; you'll still be able to deploy VM application versions past the end-of-life date.

- Exclude from latest. You can keep a version from being used as the latest version of the application.
- Target regions for replication
- Replica count per region

## Download directory

The download location of the application package and the configuration files are:

- Linux: `/var/lib/waagent/Microsoft.CPlat.Core.VMApplicationManagerLinux/<appname>/<app version>`
- Windows:
 

```
C:\Packages\Plugins\Microsoft.CPlat.Core.VMApplicationManagerWindows\1.0.4\Downloads\<appname>\<app version>
```

The install/update/remove commands should be written assuming the application package and the configuration file are in the current directory.

## File naming

When the application file gets downloaded to the VM, the file name is the same as the name you use when you create the VM application. For example, if I name my VM application `myApp`, the file that will be downloaded to the VM will also be named `myApp`, regardless of what the file name is used in the storage account. If your VM application also has a configuration file, that file is the name of the application with `_config` appended. If `myApp` has a configuration file, it will be named `myApp_config`.

For example, if I name my VM application `myApp` when I create it in the Gallery, but it's stored as `myApplication.exe` in the storage account, when it gets downloaded to the VM the file name will be `myApp`. My install string should start by renaming the file to be whatever it needs to be to run on the VM (like `myApp.exe`).

The install, update, and remove commands must be written with file naming in mind. The `configFileName` is assigned to the config file for the VM and `packageFileName` is the name assigned downloaded package on the VM. For more information regarding these additional VM settings, refer to [UserArtifactSettings](#) in our API docs.

## Command interpreter

The default command interpreters are:

- Linux: `/bin/sh`
- Windows: `cmd.exe`

It's possible to use a different interpreter like Chocolatey or PowerShell, as long as it's installed on the machine, by calling the executable and passing the command to it. For example, to have your command run in PowerShell on Windows instead of cmd, you can pass `powershell.exe -Command '<powershell command>'`

## How updates are handled

When you update an application version on a VM or VMSS, the update command you provided during deployment will be used. If the updated version doesn't have an update command, then the current version will be removed and the new version will be installed.

Update commands should be written with the expectation that it could be updating from any older version of the VM application.

## Tips for creating VM Applications on Linux

Third party applications for Linux can be packaged in a few ways. Let's explore how to handle creating the install

commands for some of the most common.

### .tar and .gz files

These are compressed archives and can be extracted to a desired location. Check the installation instructions for the original package to see if they need to be extracted to a specific location. If .tar.gz file contains source code, refer to the instructions for the package for how to install from source.

Example to install command to install `golang` on a Linux machine:

```
tar -C /usr/local -xzf go_linux
```

Example remove command:

```
rm -rf /usr/local/go
```

### .deb, .rpm, and other platform specific packages

You can download individual packages for platform specific package managers, but they usually don't contain all the dependencies. For these files, you must also include all dependencies in the application package, or have the system package manager download the dependencies through the repositories that are available to the VM. If you're working with a VM with restricted internet access, you must package all the dependencies yourself.

Figuring out the dependencies can be a bit tricky. There are third party tools that can show you the entire dependency tree.

On Ubuntu, you can run `apt-get install <name> --simulate` to show all the packages that will be installed for the `apt-get install <name>` command. Then you can use that output to download all .deb files to create an archive that can be used as the application package. The downside to this method is that it doesn't show the dependencies that are already installed on the VM.

Example, to create a VM application package to install PowerShell for Ubuntu, run the command

`apt-get install powershell --simulate` on a new Ubuntu VM. Check the output of the line **The following NEW packages will be installed** which lists the following packages:

- `liblttng-ust-ctl14`
- `liblttng-ust0`
- `liburcu6`
- `powershell`.

Download these files using `apt-get download` and create a tar archive with all files at the root level. This tar archive will be the application package file. The install command in this case is:

```
tar -xf powershell && dpkg -i ./liblttng-ust-ctl14_2.10.1-1_amd64.deb ./liburcu6_0.10.1-1ubuntu1_amd64.deb  
./liblttng-ust0_2.10.1-1_amd64.deb ./powershell_7.1.2-1.ubuntu.18.04_amd64.deb
```

And the remove command is:

```
dpkg -r powershell && apt autoremove
```

Use `apt autoremove` instead of explicitly trying to remove all the dependencies. You may have installed other applications with overlapping dependencies, and in that case, an explicit remove command would fail.

In case you don't want to resolve the dependencies yourself and apt/rpm is able to connect to the repositories, you can install an application with just one .deb/.rpm file and let apt/rpm handle the dependencies.

Example install command:

```
dpkg -i <appname> || apt --fix-broken install -y
```

## Tips for creating VM Applications on Windows

Most third party applications in Windows are available as .exe or .msi installers. Some are also available as extract and run zip files. Let us look at the best practices for each of them.

### .exe installer

Installer executables typically launch a user interface (UI) and require someone to select through the UI. If the installer supports a silent mode parameter, it should be included in your installation string.

Cmd.exe also expects executable files to have the extension .exe, so you need to rename the file to have the .exe extension.

If I wanted to create a VM application package for myApp.exe, which ships as an executable, my VM Application is called 'myApp', so I write the command assuming that the application package is in the current directory:

```
"move .\\myApp .\\myApp.exe & myApp.exe /S -config myApp_config"
```

If the installer executable file doesn't support an uninstall parameter, you can sometimes look up the registry on a test machine to know here the uninstaller is located.

In the registry, the uninstall string is stored in

```
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\<installed application name>\UninstallString
```

so I would use the contents as my remove command:

```
'\"C:\\Program Files\\myApp\\uninstall\\helper.exe\" /S'
```

### .msi installer

For command line execution of .msi installers, the commands to install or remove an application should use msieexec. Typically, msieexec runs as its own separate process and cmd doesn't wait for it to complete, which can lead to problems when installing more than one VM application. The start command can be used with msieexec to ensure that the installation completes before the command returns. For example:

```
start /wait %windir%\system32\msieexec.exe /i myapp /quiet /forcerestart /log myapp_install.log
```

Example remove command:

```
start /wait %windir%\system32\msieexec.exe /x $appname /quiet /forcerestart /log ${appname}_uninstall.log
```

### Zipped files

For .zip or other zipped files, rename and unzip the contents of the application package to the desired destination.

Example install command:

```
rename myapp myapp.zip && mkdir C:\\myapp && powershell.exe -Command "Expand-Archive -path myapp.zip -destinationpath C:\\myapp"
```

Example remove command:

```
rmdir /S /Q C:\\myapp
```

## Treat failure as deployment failure

The VM application extension always returns a *success* regardless of whether any VM app failed while being installed/updated/removed. The VM Application extension will only report the extension status as failure when there's a problem with the extension or the underlying infrastructure. This is triggered by the "treat failure as deployment failure" flag which is set to `$false` by default and can be changed to `$true`. The failure flag can be configured in [PowerShell](#) or [CLI](#).

## Troubleshooting VM Applications

To know whether a particular VM application was successfully added to the VM instance, check the message of the VM Application extension.

To learn more about getting the status of VM extensions, see [Virtual machine extensions and features for Linux](#) and [Virtual machine extensions and features for Windows](#).

To get status of VM extensions, use [Get-AzVM](#):

```
Get-AzVM -name <VM name> -ResourceGroupName <resource group name> -Status | convertto-json -Depth 10
```

To get status of scale set extensions, use [Get-AzVMSS](#):

```
$result = Get-AzVmssVM -ResourceGroupName $rgName -VMScaleSetName $vmssName -InstanceView
$resultSummary = New-Object System.Collections.ArrayList
$result | ForEach-Object {
    $res = @{
        instanceId = $_.InstanceId;
        vmappStatus = $_.InstanceView.Extensions | Where-Object {$_._Name -eq "VMAppExtension"}
    }
    $resultSummary.Add($res) | Out-Null
}
$resultSummary | convertto-json -depth 5
```

## Error messages

MESSAGE	DESCRIPTION
Current VM Application Version {name} was deprecated at {date}.	You tried to deploy a VM Application version that has already been deprecated. Try using <code>latest</code> instead of specifying a specific version.
Current VM Application Version {name} supports OS {OS}, while current OSDisk's OS is {OS}.	You tried to deploy a Linux application to Windows instance or vice versa.
The maximum number of VM applications (max=5, current={count}) has been exceeded. Use fewer applications and retry the request.	We currently only support five VM applications per VM or scale set.
More than one VM Application was specified with the same packageReferenceld.	The same application was specified more than once.

MESSAGE	DESCRIPTION
Subscription not authorized to access this image.	The subscription doesn't have access to this application version.
Storage account in the arguments doesn't exist.	There are no applications for this subscription.
The platform image {image} isn't available. Verify that all fields in the storage profile are correct. For more details about storage profile information, please refer to <a href="https://aka.ms/storageprofile">https://aka.ms/storageprofile</a> .	The application doesn't exist.
The gallery image {image} is not available in {region} region. Please contact image owner to replicate to this region, or change your requested region.	The gallery application version exists, but it was not replicated to this region.
The SAS is not valid for source uri {uri}.	A <code>Forbidden</code> error was received from storage when attempting to retrieve information about the url (either mediaLink or defaultConfigurationLink).
The blob referenced by source uri {uri} doesn't exist.	The blob provided for the mediaLink or defaultConfigurationLink properties doesn't exist.
The gallery application version url {url} cannot be accessed due to the following error: remote name not found. Ensure that the blob exists and that it's either publicly accessible or is a SAS url with read privileges.	The most likely case is that a SAS uri with read privileges was not provided.
The gallery application version url {url} cannot be accessed due to the following error: {error description}. Ensure that the blob exists and that it's either publicly accessible or is a SAS url with read privileges.	There was an issue with the storage blob provided. The error description will provide more information.
Operation {operationName} is not allowed on {application} since it is marked for deletion. You can only retry the Delete operation (or wait for an ongoing one to complete).	Attempt to update an application that's currently being deleted.
The value {value} of parameter 'galleryApplicationVersion.properties.publishingProfile.replicaCount' is out of range. The value must be between 1 and 3, inclusive.	Only between 1 and 3 replicas are allowed for VM Application versions.
Changing property 'galleryApplicationVersion.properties.publishingProfile.manageActions.install' is not allowed. (or update, delete)	It is not possible to change any of the manage actions on an existing VmApplication. A new VmApplication version must be created.
Changing property 'galleryApplicationVersion.properties.publishingProfile.settings.packageFileName' is not allowed. (or configFileName)	It is not possible to change any of the settings, such as the package file name or config file name. A new VmApplication version must be created.
The blob referenced by source uri {uri} is too big: size = {size}. The maximum blob size allowed is '1 GB'.	The maximum size for a blob referred to by mediaLink or defaultConfigurationLink is currently 1 GB.
The blob referenced by source uri {uri} is empty.	An empty blob was referenced.
{type} blob type is not supported for {operation} operation. Only page blobs and block blobs are supported.	VmApplications only supports page blobs and block blobs.

MESSAGE	DESCRIPTION
The SAS is not valid for source uri {uri}.	The SAS uri supplied for mediaLink or defaultConfigurationLink is not a valid SAS uri.
Cannot specify {region} in target regions because the subscription is missing required feature {featureName}. Either register your subscription with the required feature or remove the region from the target region list.	To use VmApplications in certain restricted regions, one must have the feature flag registered for that subscription.
Gallery image version publishing profile regions {regions} must contain the location of image version {location}.	The list of regions for replication must contain the location where the application version is.
Duplicate regions are not allowed in target publishing regions.	The publishing regions may not have duplicates.
Gallery application version resources currently do not support encryption.	The encryption property for target regions is not supported for VM Applications
Entity name doesn't match the name in the request URL.	The gallery application version specified in the request url doesn't match the one specified in the request body.
The gallery application version name is invalid. The application version name should follow Major(int32).Minor(int32).Patch(int32) format, where int is between 0 and 2,147,483,647 (inclusive). e.g. 1.0.0, 2018.12.1 etc.	The gallery application version must follow the format specified.

## Next steps

- Learn how to [create and deploy VM application packages](#).

# Create and deploy VM Applications

9/21/2022 • 12 minutes to read • [Edit Online](#)

VM Applications are a resource type in Azure Compute Gallery (formerly known as Shared Image Gallery) that simplifies management, sharing and global distribution of applications for your virtual machines.

## IMPORTANT

Deploying VM applications in Azure Compute Gallery **do not currently support using Azure policies**.

## Prerequisites

Before you get started, make sure you have the following:

This article assumes you already have an Azure Compute Gallery. If you don't already have a gallery, create one first. To learn more, see [Create a gallery for storing and sharing resources](#).

You should have uploaded your application to a container in an [Azure storage account](#). Your application can be stored in a block or page blob. If you choose to use a page blob, you need to byte align the files before you upload them. Here's a sample that will byte align your file:

```
$inputFile = <the file you want to pad>

$fileInfo = Get-Item -Path $inputFile

$remainder = $ fileInfo.Length % 512

if ($remainder -ne 0){

    $difference = 512 - $remainder

    $bytesToPad = [System.Byte[]]::CreateInstance([System.Byte], $difference)

    Add-Content -Path $inputFile -Value $bytesToPad -Encoding Byte
}
```

You need to make sure the files are publicly available, or you'll need the SAS URI for the files in your storage account. You can use [Storage Explorer](#) to quickly create a SAS URI if you don't already have one.

If you're using PowerShell, you need to be using version 3.11.0 of the Az.Storage module.

To learn more about the installation mechanism, see the [command interpreter](#).

## Create the VM application

Choose an option below for creating your VM application definition and version:

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

1. Go to the [Azure portal](#), then search for and select **Azure Compute Gallery**.

2. Select the gallery you want to use from the list.
3. On the page for your gallery, select **Add** from the top of the page and then select **VM application definition** from the drop-down. The **Create a VM application definition** page will open.
4. In the **Basics** tab, enter a name for your application and choose whether the application is for VMs running Linux or Windows.
5. Select the **Publishing options** tab if you want to specify any of the following optional settings for your VM application definition:
  - A description of the VM application definition.
  - End of life date
  - Link to a Eula
  - URI of a privacy statement
  - URI for release notes
6. When you're done, select **Review + create**.
7. When validation completes, select **Create** to have the definition deployed.
8. Once the deployment is complete, select **Go to resource**.
9. On the page for the application, select **Create a VM application version**. The **Create a VM Application Version** page will open.
10. Enter a version number like 1.0.0.
11. Select the region where you've uploaded your application package.
12. Under **Source application package**, select **Browse**. Select the storage account, then the container where your package is located. Select the package from the list and then click **Select** when you're done.  
Alternatively, you can paste the SAS URI in this field if preferred.
13. Type in the **Install script**. You can also provide the **Uninstall script** and **Update script**. See the [Overview](#) for information on how to create the scripts.
14. If you have a default configuration file uploaded to a storage account, you can select it in **Default configuration**.
15. Select **Exclude from latest** if you don't want this version to appear as the latest version when you create a VM.
16. For **End of life date**, choose a date in the future to track when this version should be retired. It isn't deleted or removed automatically, it's only for your own tracking.
17. To replicate this version to other regions, select the **Replication** tab and add more regions and make changes to the number of replicas per region. The original region where your version was created must be in the list and can't be removed.
18. When you're done making changes, select **Review + create** at the bottom of the page.
19. When validation shows as passed, select **Create** to deploy your VM application version.

Now you can create a VM and deploy the VM application to it using the portal. Just create the VM as usual, and under the **Advanced** tab, choose **Select a VM application to install**.

Add additional configuration, agents, scripts or applications via virtual machine extensions or cloud-init.

## Extensions

Extensions provide post-deployment configuration and automation.

Extensions [\(1\)](#)

Select an extension to install

## VM applications (preview)

VM applications contain application files that are securely and reliably downloaded on your VM after deployment. In addition to the application files, an install and uninstall script are included in the application. You can easily add or remove applications on your VM after create. [Learn more ↗](#)

Select a VM application to install

Select the VM application from the list, and then select **Save** at the bottom of the page.

The screenshot shows a 'Select VM applications' dialog box. At the top, there's a search bar and a 'Showing 1 application' message. Below is a table with columns: Application name, Description, Gallery name, Resource group, Operating system type, and Version. One row is visible for 'myApp' from 'myGallery' with version 1.0.0. A red box highlights the checkbox next to 'myApp'. At the bottom are 'Save' and 'Cancel' buttons.

If you've more than one VM application to install, you can set the install order for each VM application back on the **Advanced** tab.

You can also deploy the VM application to currently running VMs. Select the **Extensions + applications** option under **Settings** in the left menu when viewing the VM details in the portal.

Choose **VM applications** and then select **Add application** to add your VM application.

Home > myVM >

The screenshot shows the 'myVM | Extensions + applications' settings page. The 'VM Applications' tab is active. On the right, there's a table with 'Install order' and 'Application name' columns. A new row is being added with a grey placeholder bar. On the left, a sidebar lists various settings: Networking, Connect, Disks, Size, Security, Advisor recommendations, Extensions + applications (selected), and Continuous delivery. A search bar is at the top left.

Select the VM application from the list, and then select **Save** at the bottom of the page.

To show the VM application status, go to the Extensions + applications tab/settings and check the status of the VMAppExtension:

To show the VM application status for VMSS, go to the VMSS page, Instances, select one of them, then go to VMAppExtension:

## Next steps

Learn more about [VM applications](#).

# Find Azure Marketplace image information using the Azure CLI

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This topic describes how to use the Azure CLI to find VM images in the Azure Marketplace. Use this information to specify a Marketplace image when you create a VM programmatically with the CLI, Resource Manager templates, or other tools.

You can also browse available images and offers using the [Azure Marketplace](#) or [Azure PowerShell](#).

## Terminology

A Marketplace image in Azure has the following attributes:

- **Publisher:** The organization that created the image. Examples: Canonical, MicrosoftWindowsServer
- **Offer:** The name of a group of related images created by a publisher. Examples: UbuntuServer, WindowsServer
- **SKU:** An instance of an offer, such as a major release of a distribution. Examples: 18.04-LTS, 2019-Datacenter
- **Version:** The version number of an image SKU.

These values can be passed individually or as an image *URN*, combining the values separated by the colon (:). For example: *Publisher.Offer.Sku.Version*. You can replace the version number in the URN with `latest` to use the latest version of the image.

If the image publisher provides additional license and purchase terms, then you must accept those before you can use the image. For more information, see [Check the purchase plan information](#).

## List popular images

Run the `az vm image list` command, without the `--all` option, to see a list of popular VM images in the Azure Marketplace. For example, run the following command to display a cached list of popular images in table format:

```
az vm image list --output table
```

The output includes the image URN. You can also use the *UrnAlias* which is a shortened version created for popular images like *UbuntuLTS*.

Offer UrnAlias	Publisher Version	Sku	Urn
CentOS	OpenLogic	7.5	OpenLogic:CentOS:7.5:latest
CentOS		latest	
CoreOS	CoreOS	Stable	CoreOS:CoreOS:Stable:latest
CoreOS		latest	
debian-10	Debian	10	Debian:debian-10:10:latest
Debian		latest	
openSUSE-Leap	SUSE	42.3	SUSE:openSUSE-Leap:42.3:latest
openSUSE-Leap		latest	
RHEL	RedHat	7-LVM	RedHat:RHEL:7-LVM:latest
RHEL		latest	
SLES	SUSE	15	SUSE:SLES:15:latest
SLES		latest	
UbuntuServer	Canonical	18.04-LTS	Canonical:UbuntuServer:18.04-LTS:latest
UbuntuLTS		latest	
WindowsServer	MicrosoftWindowsServer	2019-Datacenter	MicrosoftWindowsServer:WindowsServer:2019-
Datacenter:latest	Win2019Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2016-Datacenter	MicrosoftWindowsServer:WindowsServer:2016-
Datacenter:latest	Win2016Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2012-R2-Datacenter	MicrosoftWindowsServer:WindowsServer:2012-R2-
Datacenter:latest	Win2012R2Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2012-Datacenter	MicrosoftWindowsServer:WindowsServer:2012-
Datacenter:latest	Win2012Datacenter	latest	
WindowsServer	MicrosoftWindowsServer	2008-R2-SP1	MicrosoftWindowsServer:WindowsServer:2008-R2-
SP1:latest	Win2008R2SP1	latest	

## Find specific images

To find a specific VM image in the Marketplace, use the `az vm image list` command with the `--all` option. This version of the command takes some time to complete and can return lengthy output, so you usually filter the list by `--publisher` or another parameter.

For example, the following command displays all Debian offers (remember that without the `--all` switch, it only searches the local cache of common images):

```
az vm image list --offer Debian --all --output table
```

Partial output:

Offer	Publisher	SKU
Urn		
Version		
-----	-----	-----
-----	-----	-----
apache-solr-on-debian	apps-4-rent	apache-solr-on-debian
apps-4-rent:apache-solr-on-debian:apache-solr-on-debian:1.0.0		1.0.0
atomized-h-debian10-v1	atomizedinc1587939464368	hdebian10plan
atomizedinc1587939464368:atomized-h-debian10-v1:hdebian10plan:1.0.0		1.0.0
atomized-h-debian9-v1	atomizedinc1587939464368	hdebian9plan
atomizedinc1587939464368:atomized-h-debian9-v1:hdebian9plan:1.0.0		1.0.0
atomized-r-debian10-v1	atomizedinc1587939464368	rdebian10plan
atomizedinc1587939464368:atomized-r-debian10-v1:rdebian10plan:1.0.0		1.0.0
atomized-r-debian9-v1	atomizedinc1587939464368	rdebian9plan
atomizedinc1587939464368:atomized-r-debian9-v1:rdebian9plan:1.0.0		1.0.0
cis-debian-linux-10-11	center-for-internet-security-inc	cis-debian10-11
center-for-internet-security-inc:cis-debian-linux-10-11:cis-debian10-11:1.0.7		1.0.7
cis-debian-linux-10-11	center-for-internet-security-inc	cis-debian10-11
center-for-internet-security-inc:cis-debian-linux-10-11:cis-debian10-11:1.0.8		1.0.8
cis-debian-linux-10-11	center-for-internet-security-inc	cis-debian10-11
center-for-internet-security-inc:cis-debian-linux-10-11:cis-debian10-11:1.0.9		1.0.9
cis-debian-linux-9-11	center-for-internet-security-inc	cis-debian9-11
center-for-internet-security-inc:cis-debian-linux-9-11:cis-debian9-11:1.0.18		1.0.18
cis-debian-linux-9-11	center-for-internet-security-inc	cis-debian9-11
center-for-internet-security-inc:cis-debian-linux-9-11:cis-debian9-11:1.0.19		1.0.19
cis-debian-linux-9-11	center-for-internet-security-inc	cis-debian9-11
center-for-internet-security-inc:cis-debian-linux-9-11:cis-debian9-11:1.0.20		1.0.20
apache-web-server-with-debian-10	cognosys	apache-web-server-with-debian-10
cognosys:apache-web-server-with-debian-10:apache-web-server-with-debian-10:1.2019.1008		
1.2019.1008		
docker-ce-with-debian-10	cognosys	docker-ce-with-debian-10
cognosys:docker-ce-with-debian-10:docker-ce-with-debian-10:1.2019.0710		
1.2019.0710		
Debian	credativ	8
credativ:Debian:8:8.0.201602010		
8.0.201602010		
Debian	credativ	8
credativ:Debian:8:8.0.201603020		
8.0.201603020		
Debian	credativ	8
credativ:Debian:8:8.0.201604050		
8.0.201604050		
...		

## Look at all available images

Another way to find an image in a location is to run the [az vm image list-publishers](#), [az vm image list-offers](#), and [az vm image list-skus](#) commands in sequence. With these commands, you determine these values:

1. List the image publishers for a location. In this example, we are looking at the *West US* region.

```
az vm image list-publishers --location westus --output table
```

2. For a given publisher, list their offers. In this example, we add *Canonical* as the publisher.

```
az vm image list-offers --location westus --publisher Canonical --output table
```

3. For a given offer, list their SKUs. In this example, we add *UbuntuServer* as the offer.

```
az vm image list-skus --location westus --publisher Canonical --offer UbuntuServer --output table
```

4. For a given publisher, offer, and SKU, show all of the versions of the image. In this example, we add *18.04-LTS* as the SKU.

```
az vm image list \
--location westus \
--publisher Canonical \
--offer UbuntuServer \
--sku 18.04-LTS \
--all --output table
```

Pass this value of the URN column with the `--image` parameter when you create a VM with the [az vm create](#) command. You can also replace the version number in the URN with "latest", to simply use the latest version of the image.

If you deploy a VM with a Resource Manager template, you set the image parameters individually in the `imageReference` properties. See the [template reference](#).

## Check the purchase plan information

Some VM images in the Azure Marketplace have additional license and purchase terms that you must accept before you can deploy them programmatically.

To deploy a VM from such an image, you'll need to accept the image's terms the first time you use it, once per subscription. You'll also need to specify *purchase plan* parameters to deploy a VM from that image.

To view an image's purchase plan information, run the [az vm image show](#) command with the URN of the image. If the `plan` property in the output is not `null`, the image has terms you need to accept before programmatic deployment.

For example, the Canonical Ubuntu Server 18.04 LTS image doesn't have additional terms, because the `plan` information is `null`:

```
az vm image show --location westus --urn Canonical:UbuntuServer:18.04-LTS:latest
```

Output:

```
{
  "dataDiskImages": [],
  "id": "/Subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/Providers/Microsoft.Compute/Locations/westus/Publishers/Canonical/ArtifactTypes/VMImage/Offers/
UbuntuServer/Skus/18.04-LTS/Versions/18.04.201901220",
  "location": "westus",
  "name": "18.04.201901220",
  "osDiskImage": {
    "operatingSystem": "Linux"
  },
  "plan": null,
  "tags": null
}
```

Running a similar command for the RabbitMQ Certified by Bitnami image shows the following `plan` properties: `name`, `product`, and `publisher`. (Some images also have a `promotion code` property.)

```
az vm image show --location westus --urn bitnami:rabbitmq:rabbitmq:latest
```

Output:

```
{  
    "dataDiskImages": [],  
    "id": "/Subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxxxxx/Providers/Microsoft.Compute/Locations/westus/Publishers/bitnami/ArtifactTypes/VMImage/Offers/ra  
bbitmq/Skus/rabbitmq/Versions/3.7.1901151016",  
    "location": "westus",  
    "name": "3.7.1901151016",  
    "osDiskImage": {  
        "operatingSystem": "Linux"  
    },  
    "plan": {  
        "name": "rabbitmq",  
        "product": "rabbitmq",  
        "publisher": "bitnami"  
    },  
    "tags": null  
}
```

To deploy this image, you need to accept the terms and provide the purchase plan parameters when you deploy a VM using that image.

## Accept the terms

To view and accept the license terms, use the [az vm image terms](#) command. When you accept the terms, you enable programmatic deployment in your subscription. You only need to accept terms once per subscription for the image. For example:

```
az vm image terms show --urn bitnami:rabbitmq:rabbitmq:latest
```

The output includes a `licenseTextLink` to the license terms, and indicates that the value of `accepted` is `true`:

```
{  
    "accepted": true,  
    "additionalProperties": {},  
    "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxxxx/providers/Microsoft.MarketplaceOrdering/offertypes/bitnami/offers/rabbitmq/plans/rabbitmq",  
    "licenseTextLink":  
        "https://storelegalterms.blob.core.windows.net/legalterms/3E5ED_legalterms_BITNAMI%253a24RABBITMQ%253a24RABB  
ITMQ%253a24IGRT7HHPIFOBV3IQYJHEN202FGUVXXZ3WUYIMEIVF3KCUNJ7GTVXNNM23I567GBMNDWRFOY4WXJPNSPUYXNKB2QLAKCHP4IE5  
GO3B2I.txt",  
    "name": "rabbitmq",  
    "plan": "rabbitmq",  
    "privacyPolicyLink": "https://bitnami.com/privacy",  
    "product": "rabbitmq",  
    "publisher": "bitnami",  
    "retrieveDatetime": "2019-01-25T20:37:49.937096Z",  
    "signature":  
        "XXXXXXXXLAZIK7ZL2YRV5JYQXONPV76NQJW3FKMDZYCRGXZYVDGX6BVY45J03BXVMNA2COBOEYG2N0760NORU7ITTRHGZDYNJNXXXXXX",  
    "type": "Microsoft.MarketplaceOrdering/offertypes"  
}
```

To accept the terms, type:

```
az vm image terms accept --urn bitnami:rabbitmq:rabbitmq:latest
```

## Deploy a new VM using the image parameters

With information about the image, you can deploy it using the `az vm create` command.

To deploy an image that does not have plan information, like the latest Ubuntu Server 18.04 image from Canonical, pass the URN for `--image`:

```
az group create --name myURNVM --location westus
az vm create \
    --resource-group myURNVM \
    --name myVM \
    --admin-username azureuser \
    --generate-ssh-keys \
    --image Canonical:UbuntuServer:18.04-LTS:latest
```

For an image with purchase plan parameters, like the RabbitMQ Certified by Bitnami image, you pass the URN for `--image` and also provide the purchase plan parameters:

```
az group create --name myPurchasePlanRG --location westus

az vm create \
    --resource-group myPurchasePlanRG \
    --name myVM \
    --admin-username azureuser \
    --generate-ssh-keys \
    --image bitnami:rabbitmq:rabbitmq:latest \
    --plan-name rabbitmq \
    --plan-product rabbitmq \
    --plan-publisher bitnami
```

If you get a message about accepting the terms of the image, review section [Accept the terms](#). Make sure the output of `az vm image accept-terms` returns the value `"accepted": true,` showing that you have accepted the terms of the image.

## Using an existing VHD with purchase plan information

If you have an existing VHD from a VM that was created using a paid Azure Marketplace image, you might need to supply the purchase plan information when you create a new VM from that VHD.

If you still have the original VM, or another VM created using the same marketplace image, you can get the plan name, publisher, and product information from it using `az vm get-instance-view`. This example gets a VM named `myVM` in the `myResourceGroup` resource group and then displays the purchase plan information.

```
az vm get-instance-view -g myResourceGroup -n myVM --query plan
```

If you didn't get the plan information before the original VM was deleted, you can file a [support request](#). They will need the VM name, subscription ID and the time stamp of the delete operation.

Once you have the plan information, you can create the new VM using the `--attach-os-disk` parameter to specify the VHD.

```
az vm create \
--resource-group myResourceGroup \
--name myNewVM \
--nics myNic \
--size Standard_DS1_v2 --os-type Linux \
--attach-os-disk myVHD \
--plan-name planName \
--plan-publisher planPublisher \
--plan-product planProduct
```

## Next steps

To create a virtual machine quickly by using the image information, see [Create and Manage Linux VMs with the Azure CLI](#).

# Find and use Azure Marketplace VM images with Azure PowerShell

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes how to use Azure PowerShell to find VM images in the Azure Marketplace. You can then specify a Marketplace image and plan information when you create a VM.

You can also browse available images and offers using the [Azure Marketplace](#) or the [Azure CLI](#).

## Terminology

A Marketplace image in Azure has the following attributes:

- **Publisher:** The organization that created the image. Examples: Canonical, MicrosoftWindowsServer
- **Offer:** The name of a group of related images created by a publisher. Examples: UbuntuServer, WindowsServer
- **SKU:** An instance of an offer, such as a major release of a distribution. Examples: 18.04-LTS, 2019-Datacenter
- **Version:** The version number of an image SKU.

These values can be passed individually or as an image *URN*, combining the values separated by the colon (:). For example: *Publisher.Offer.Sku.Version*. You can replace the version number in the URN with `latest` to use the latest version of the image.

If the image publisher provides additional license and purchase terms, then you must accept those before you can use the image. For more information, see [Accept purchase plan terms](#).

## List images

You can use PowerShell to narrow down a list of images. Replace the values of the variables to meet your needs.

1. List the image publishers using [Get-AzVMImagePublisher](#).

```
$locName=<location>
Get-AzVMImagePublisher -Location $locName | Select PublisherName
```

2. For a given publisher, list their offers using [Get-AzVMImageOffer](#).

```
$pubName=<publisher>
Get-AzVMImageOffer -Location $locName -PublisherName $pubName | Select Offer
```

3. For a given publisher and offer, list the SKUs available using [Get-AzVMImageSku](#).

```
$offerName=<offer>
Get-AzVMImageSku -Location $locName -PublisherName $pubName -Offer $offerName | Select Skus
```

4. For a SKU, list the versions of the image using [Get-AzVMImage](#).

```
$skuName="<SKU>"  
Get-AzVMImage -Location $locName -PublisherName $pubName -Offer $offerName -Sku $skuName | Select  
Version
```

You can also use `latest` if you want to use the latest image and not a specific older version.

Now you can combine the selected publisher, offer, SKU, and version into a URN (values separated by `:`). Pass this URN with the `-Image` parameter when you create a VM with the [New-AzVM](#) cmdlet. You can also replace the version number in the URN with `latest` to get the latest version of the image.

If you deploy a VM with a Resource Manager template, then you'll set the image parameters individually in the `imageReference` properties. See the [template reference](#).

## View purchase plan properties

Some VM images in the Azure Marketplace have additional license and purchase terms that you must accept before you can deploy them programmatically. You'll need to accept the image's terms once per subscription.

To view an image's purchase plan information, run the `Get-AzVMImage` cmdlet. If the `PurchasePlan` property in the output is not `null`, the image has terms you need to accept before programmatic deployment.

For example, the *Windows Server 2016 Datacenter* image doesn't have additional terms, so the `PurchasePlan` information is `null`:

```
$version = "2016.127.20170406"  
Get-AzVMImage -Location $locName -PublisherName $pubName -Offer $offerName -Skus $skuName -Version $version
```

The output will look similar to the following:

```
Id : /Subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/Providers/Microsoft.Compute/Locations/westus/Publishers/MicrosoftWindowsServer/ArtifactTypes/VM  
Image/Offers/WindowsServer/Skus/2016-Datacenter/Versions/2019.0.20190115  
Location : westus  
PublisherName : MicrosoftWindowsServer  
Offer : WindowsServer  
Skus : 2019-Datacenter  
Version : 2019.0.20190115  
FilterExpression :  
Name : 2019.0.20190115  
OSDiskImage : {  
    "operatingSystem": "Windows"  
}  
PurchasePlan : null  
DataDiskImages : []
```

The example below shows a similar command for the *Data Science Virtual Machine - Windows 2016* image, which has the following `PurchasePlan` properties: `name`, `product`, and `publisher`. Some images also have a `promotion code` property. To deploy this image, see the following sections to accept the terms and to enable programmatic deployment.

```
Get-AzVMImage -Location "westus" -PublisherName "microsoft-ads" -Offer "windows-data-science-vm" -Skus  
"windows2016" -Version "0.2.02"
```

The output will look similar to the following:

```

Id          : /Subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxxx/Providers/Microsoft.Compute/Locations/westus/Publishers/microsoft-
ads/ArtifactTypes/VMImage/Offers/windows-data-science-vm/Skus/windows2016/Versions/19.01.14
Location    : westus
PublisherName : microsoft-ads
Offer        : windows-data-science-vm
Skus         : windows2016
Version      : 19.01.14
FilterExpression :
Name         : 19.01.14
OSDiskImage   : {
                  "operatingSystem": "Windows"
                }
PurchasePlan   : {
                  "publisher": "microsoft-ads",
                  "name": "windows2016",
                  "product": "windows-data-science-vm"
                }
DataDiskImages : []

```

To view the license terms, use the [Get-AzMarketplaceterms](#) cmdlet and pass in the purchase plan parameters. The output provides a link to the terms for the Marketplace image and shows whether you previously accepted the terms. Be sure to use all lowercase letters in the parameter values.

```
Get-AzMarketplaceterms -Publisher "microsoft-ads" -Product "windows-data-science-vm" -Name "windows2016"
```

The output will look similar to the following:

```

Publisher      : microsoft-ads
Product        : windows-data-science-vm
Plan           : windows2016
LicenseTextLink :
https://storelegalterms.blob.core.windows.net/legalterms/3E5ED_legalterms_MICROSOFT%253a2DADS%253a24WINDOWS%
253a2DDATA%253a2DSCIENCE%253a2DVM%253a24WINDOWS2016%253a240C5SKMQOXSED66BBSNTF4XRC54XLOHP7QMPV54DQU7JCBZWYFP
35IDPOWTUXUC7ZAG7W6ZMDD6NHWNKUIVSYBUTZ245F44SU5AD7Q.txt
PrivacyPolicyLink : https://www.microsoft.com/EN-US/privacystatement/OnlineServices/Default.aspx
Signature      :
2UWH6PHSAIM4U22HXPXW25AL2NHUJ7Y7GRV27EBL6SUIDURGMYG6IID03P47FFIBBDHZHSQTR7PNK6VIIRYJRQ3WXSE6BTNUNENXA
Accepted       : False
Signdate       : 1/25/2019 7:43:00 PM

```

## Accept purchase plan terms

Use the [Set-AzMarketplaceterms](#) cmdlet to accept or reject the terms. You only need to accept terms once per subscription for the image. Be sure to use all lowercase letters in the parameter values.

```

$agreementTerms=Get-AzMarketplaceterms -Publisher "microsoft-ads" -Product "windows-data-science-vm" -Name
"windows2016"

Set-AzMarketplaceTerms -Publisher "microsoft-ads" -Product "windows-data-science-vm" -Name "windows2016" -
Terms $agreementTerms -Accept

```

```
Publisher      : microsoft-ads
Product       : windows-data-science-vm
Plan          : windows2016
LicenseTextLink :
https://storelegalterms.blob.core.windows.net/legalterms/3E5ED_legalterms_MICROSOFT%253a2DADS%253a24WINDOWS%
253a2DDATA%253a2DSCIENCE%253a2DV

M%253a24WINDOWS2016%253a240C5SKMQ0XSED66BBSNTF4XRCS4XLOHP7QMPV54DQU7JCBZWYFP35IDPOWTUKXUC7ZAG7W6ZMDD6NHWNKUI
VSYBZUTZ245F44SU5AD7Q.txt
PrivacyPolicyLink : https://www.microsoft.com/EN-US/privacystatement/OnlineServices/Default.aspx
Signature      :
XXXXXXXX3MNJ5SROEG2BYDA2YGEPU33GXTD3UFPLPC4BAVKAUL3PDYL3KBKBLG4ZCDJZVNSA7KJWTGMDSYDD6KRLV3LV274DLBXXXXXX
Accepted       : True
Signdate       : 2/23/2018 7:49:31 PM
```

## Create a new VM from a marketplace image

If you already have the information about what image you want to use, you can pass that information into [Set-AzVMSourceImage](#) cmdlet to add image information to the VM configuration. See the next sections for searching and listing the images available in the marketplace.

Some paid images also require that you provide purchase plan information using the [Set-AzVMPlan](#).

```
...

$vmConfig = New-AzVMConfig -VMName "myVM" -VMSize Standard_D1

# Set the Marketplace image
$offerName = "windows-data-science-vm"
$skuName = "windows2016"
$version = "19.01.14"
$vmConfig = Set-AzVMSourceImage -VM $vmConfig -PublisherName $publisherName -Offer $offerName -Skus $skuName
-Version $version

# Set the Marketplace plan information, if needed
$publisherName = "microsoft-ads"
$productName = "windows-data-science-vm"
$planName = "windows2016"
$vmConfig = Set-AzVMPlan -VM $vmConfig -Publisher $publisherName -Product $productName -Name $planName

...
```

You'll then pass the VM configuration along with the other configuration objects to the [New-AzVM](#) cmdlet. For a detailed example of using a VM configuration with PowerShell, see this [script](#).

If you get a message about accepting the terms of the image, see the earlier section [Accept purchase plan terms](#).

## Create a new VM from a VHD with purchase plan information

If you have an existing VHD that was created using an Azure Marketplace image, you might need to supply the purchase plan information when you create a new VM from that VHD.

If you still have the original VM, or another VM created from the same image, you can get the plan name, publisher, and product information from it using Get-AzVM. This example gets a VM named *myVM* in the *myResourceGroup* resource group and then displays the purchase plan information.

```
$vm = Get-azvm ` 
    -ResourceGroupName myResourceGroup ` 
    -Name myVM
$vm.Plan
```

If you didn't get the plan information before the original VM was deleted, you can file a [support request](#). They will need the VM name, subscription ID and the time stamp of the delete operation.

To create a VM using a VHD, refer to this article [Create a VM from a specialized VHD](#) and add in a line to add the plan information to the VM configuration using [Set-AzVMPlan](#) similar to the following:

```
$vmConfig = Set-AzVMPlan ` 
    -VM $vmConfig ` 
    -Publisher "publisherName" ` 
    -Product "productName" ` 
    -Name "planName"
```

## Next steps

To create a virtual machine quickly with the `New-AzVM` cmdlet by using basic image information, see [Create a Windows virtual machine with PowerShell](#).

For more information on using Azure Marketplace images to create custom images in an Azure Compute Gallery (formerly known as Shared Image Gallery), see [Supply Azure Marketplace purchase plan information when creating images](#).

# Use Windows client in Azure for dev/test scenarios

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

You can use Windows 7, Windows 8, or Windows 10 Enterprise (x64) in Azure for dev/test scenarios provided you have an appropriate Visual Studio (formerly MSDN) subscription.

To run Windows 10 in a production environment see, [How to deploy Windows 10 on Azure with Multitenant Hosting Rights](#).

## Subscription eligibility

Active Visual Studio subscribers (people who have acquired a Visual Studio subscription license) can use Windows client images for development and testing purposes. Windows client images can be used on your own hardware or on Azure virtual machines.

Certain Windows client images are available from the Azure Marketplace. Visual Studio subscribers within any type of offer can also [prepare and create](#) 64-bit Windows 7, Windows 8, or Windows 10 images and then [upload to Azure](#).

## Eligible offers and client images

The following table details the offer IDs that are eligible to deploy Windows client images through the Azure Marketplace. The Windows client images are only visible to the following offers.

### NOTE

Image offers are under **Windows Client** in the Azure Marketplace. Use **Windows Client** when searching for client images available to Visual Studio subscribers. If you need to purchase a Visual Studio subscription, see the various options at [Buy Visual Studio](#)

OFFER NAME	OFFER NUMBER	AVAILABLE CLIENT IMAGES
<a href="#">Pay-As-You-Go Dev/Test</a>	0023P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)
<a href="#">Visual Studio Enterprise (MPN) subscribers</a>	0029P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)
<a href="#">Visual Studio Professional subscribers</a>	0059P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)
<a href="#">Visual Studio Test Professional subscribers</a>	0060P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)

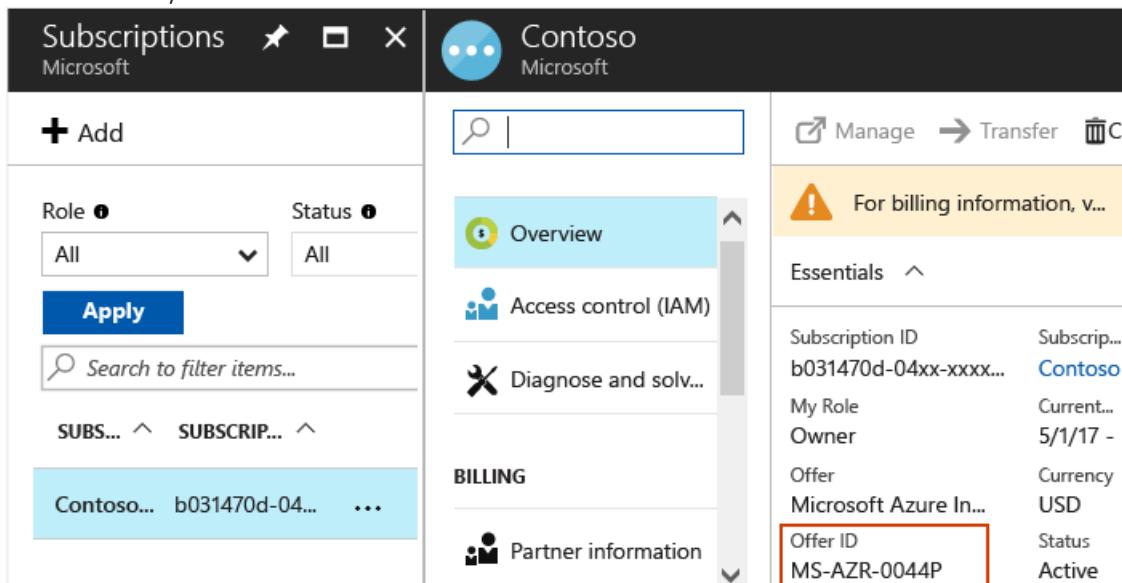
OFFER NAME	OFFER NUMBER	AVAILABLE CLIENT IMAGES
Visual Studio Enterprise subscribers	0063P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)
Visual Studio Enterprise (BizSpark) subscribers	0064P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)
Enterprise Dev/Test	0148P	Windows 10 Enterprise N (x64) Windows 8.1 Enterprise N (x64) Windows 7 Enterprise N with SP1 (x64)

For more information, see [Understand Microsoft offer types](#)

## Check your Azure subscription

If you do not know your offer ID, you can obtain it through the Azure portal.

- On the *Subscriptions* window:



The screenshot shows the Azure Subscriptions window for the Contoso Microsoft account. The left sidebar has filters for Role (All) and Status (All), with an 'Apply' button. Below that are 'SUBS...' and 'SUBSCRIP...' dropdowns, and a list item for 'Contoso... b031470d-04... ...'. The main area has tabs for 'Overview' (which is selected and highlighted in blue), 'Access control (IAM)', 'Diagnose and solv...', 'BILLING', and 'Partner information'. The 'BILLING' tab is expanded, showing a table with columns for Subscription ID, Offer, and Partner information. The 'Offer ID' row is highlighted with a red box, showing the value 'MS-AZR-0044P'.

Subscription ID	Offer	Partner information
b031470d-04xx-xxxx...	Microsoft Azure In...	MS-AZR-0044P

- Or, click **Billing** and then click your subscription ID. The offer ID appears in the *Billing* window.
- You can also view the offer ID from the '[Subscriptions](#)' tab of the Azure Account portal:

**ACCOUNT ADMINISTRATOR**

**SUBSCRIPTION ID**

**ORDER ID**

**OFFER**

Visual Studio Enterprise

**OFFER ID**

MS-AZR-0063P

**CURRENCY**

USD

**STATUS**

Active

## Next steps

You can now deploy your VMs using [PowerShell](#), [Resource Manager templates](#), or [Visual Studio](#).

# How to deploy Windows 10 on Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

For customers with Windows 10 Enterprise E3/E5 per user or Azure Virtual Desktop Access per user (User Subscription Licenses or Add-on User Subscription Licenses), Multitenant Hosting Rights for Windows 10 allows you to bring your Windows 10 Licenses to the cloud and run Windows 10 Virtual Machines on Azure without paying for another license. Multitenant Hosting Rights are only available for Windows 10 (version 1703 or later).

For more information, see [Multitenant Hosting for Windows 10](#).

## NOTE

- To use Windows 7, 8.1 and 10 images for development or testing see [Windows client in Azure for dev/test scenarios](#)
- For Windows Server licensing benefits, please refer to [Azure Hybrid use benefits for Windows Server images](#).

## Subscription Licenses that qualify for Multitenant Hosting Rights

For more details about subscription licenses that qualify to run Windows 10 on Azure, download the [Windows 10 licensing brief for Virtual Desktops](#)

## IMPORTANT

Users **must** have one of the below subscription licenses in order to use Windows 10 images in Azure for any production workload. If you do not have one of these subscription licenses, they can be purchased through your [Cloud Service Partner](#) or directly through [Microsoft](#).

## Operating systems and licenses

You have a choice of operating systems that you can use for session hosts to provide virtual desktops and remote apps. You can use different operating systems with different host pools to provide flexibility to your users. Supported dates are inline with the [Microsoft Lifecycle Policy](#). We support the following 64-bit versions of these operating systems:

### Operating system licenses

- Windows 11 Enterprise multi-session
- Windows 11 Enterprise
- Windows 10 Enterprise, version 1909 and later

### License entitlement

- Microsoft 365 E3, E5, A3, A5, F3, Business Premium, Student Use Benefit
- Windows Enterprise E3, E5
- Windows VDA E3, E5
- Windows Education A3, A5

External users can use [per-user access pricing](#) instead of license entitlement.

## Deploying Windows 10 Image from Azure Marketplace

For PowerShell, CLI and Azure Resource Manager template deployments, Windows 10 images can be found using the `PublisherName: MicrosoftWindowsDesktop` and `Offer: Windows-10`. Windows 10 version Creators Update (1809) or later is supported for Multitenant Hosting Rights.

```
Get-AzVmImageSku -Location '$location' -PublisherName 'MicrosoftWindowsDesktop' -Offer 'Windows-10'

Skus          Offer      PublisherName      Location
----          -----      -----            -----
rs4-pro       Windows-10 MicrosoftWindowsDesktop eastus
rs4-pron      Windows-10 MicrosoftWindowsDesktop eastus
rs5-enterprise Windows-10 MicrosoftWindowsDesktop eastus
rs5-enterprisen Windows-10 MicrosoftWindowsDesktop eastus
rs5-pron      Windows-10 MicrosoftWindowsDesktop eastus
```

For more information on available images see [Find and use Azure Marketplace VM images with Azure PowerShell](#)

## Uploading Windows 10 VHD to Azure

If you are uploading a generalized Windows 10 VHD, please note Windows 10 does not have built-in administrator account enabled by default. To enable the built-in administrator account, include the following command as part of the Custom Script extension.

```
Net user <username> /active:yes
```

The following PowerShell snippet is to mark all administrator accounts as active, including the built-in administrator. This example is useful if the built-in administrator username is unknown.

```
$adminAccount = Get-WmiObject Win32_UserAccount -filter "LocalAccount=True" | ? {$_.SID -Like "S-1-5-21-*-500"}
if($adminAccount.Disabled)
{
    $adminAccount.Disabled = $false
    $adminAccount.Put()
}
```

For more information:

- [How to upload VHD to Azure](#)
- [How to prepare a Windows VHD to upload to Azure](#)

## Deploying Windows 10 with Multitenant Hosting Rights

Make sure you have [installed and configured the latest Azure PowerShell](#). Once you have prepared your VHD, upload the VHD to your Azure Storage account using the `Add-AzVhd` cmdlet as follows:

```
Add-AzVhd -ResourceGroupName "myResourceGroup" -LocalFilePath "C:\Path\To\myvhd.vhd" ` 
-Destination "https://mystorageaccount.blob.core.windows.net/vhds/myvhd.vhd"
```

**Deploy using Azure Resource Manager Template Deployment** Within your Resource Manager templates, an additional parameter for `licenseType` can be specified. You can read more about [authoring Azure Resource Manager templates](#). Once you have your VHD uploaded to Azure, edit your Resource Manager template to include the license type as part of the compute provider and deploy your template as normal:

```
"properties": {  
    "licenseType": "Windows_Client",  
    "hardwareProfile": {  
        "vmSize": "[variables('vmSize')]"  
    }  
}
```

**Deploy via PowerShell** When deploying your Windows Server VM via PowerShell, you have an additional parameter for `-LicenseType`. Once you have your VHD uploaded to Azure, you create a VM using `New-AzVM` and specify the licensing type as follows:

```
New-AzVM -ResourceGroupName "myResourceGroup" -Location "West US" -VM $vm -LicenseType "Windows_Client"
```

## Verify your VM is utilizing the licensing benefit

Once you have deployed your VM through either the PowerShell or Resource Manager deployment method, verify the license type with `Get-AzVM` as follows:

```
Get-AzVM -ResourceGroup "myResourceGroup" -Name "myVM"
```

The output is similar to the following example for Windows 10 with correct license type:

```
Type          : Microsoft.Compute/virtualMachines  
Location     : westus  
LicenseType  : Windows_Client
```

This output contrasts with the following VM deployed without Azure Hybrid Use Benefit licensing, such as a VM deployed straight from the Azure Gallery:

```
Type          : Microsoft.Compute/virtualMachines  
Location     : westus  
LicenseType  :
```

## Additional Information about joining Azure Active Directory

Azure provisions all Windows VMs with built-in administrator account, which cannot be used to join Azure Active Directory. For example, *Settings > Account > Access Work or School > +Connect* will not work. You must create and log on as a second administrator account to join Azure AD manually. You can also configure Azure AD using a provisioning package, use the link in the *Next Steps* section to learn more.

## Next Steps

- Learn more about [Configuring VDA for Windows 10](#)
- Learn more about [Multitenant Hosting for Windows 10](#)

# Bringing and creating Linux images in Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

This overview covers the basic concepts around imaging and how to successfully build and use Linux images in Azure. Before you bring a custom image to Azure, you need to be aware of the types and options available to you.

This article will talk through the image decision points and requirements as well as explain key concepts so that you can follow this and be able to create your own custom images to your specification.

## Difference between managed disks and images

Azure allows you to bring a VHD to the platform to use as a [Managed Disk](#) or as a source for an image.

An Azure managed disk is a single VHD. You can either take an existing VHD and create a managed disk from it, or create an empty managed disk from scratch. You can create VMs from managed disks by attaching the disk to the VM, but you can only use a VHD with one VM. You won't be able to modify any OS properties as Azure will just try to turn on the VM and start up using that disk.

Azure images can be made up of multiple OS disks and data disks. When you use a managed image to create a VM, the platform makes a copy of the image and uses that to create the VM. This allows managed images to support reusing the same image for multiple VMs. Azure also provides advanced management capabilities for images, like global replication and versioning through [Azure Compute Gallery](#) (formerly known as Shared Image Gallery).

## Generalized and specialized

Azure offers two main image types, generalized and specialized. The terms generalized and specialized are originally Windows terms which migrated in to Azure. These types define how the platform will handle the VM when it turns it on. Both types have advantages, disadvantages, and prerequisites. Before you get started, you need to know what image type you will need. Below summarizes the scenarios and type you would need to choose:

SCENARIO	IMAGE TYPE	STORAGE OPTIONS
Create an image that can be configured for use by multiple VMs. You can set the hostname, add an admin user, and perform other tasks during first boot.	Generalized	Azure Compute Gallery or stand-alone managed images
Create an image from a VM snapshot or a backup.	Specialized	Azure Compute Gallery or a managed disk
Quickly create an image that does not need any configuration for creating multiple VMs.	Specialized	Azure Compute Gallery

### Generalized images

A generalized image is an image that requires setup to be completed on first boot. For example, on first boot

you set the hostname, admin user, and other VM-specific configurations. This is useful when you want the image to be reused multiple times and when you want to pass in parameters during creation. If the generalized image contains the Azure agent, the agent will process the parameters and signal back to the platform that the initial configuration has completed. This process is called [provisioning](#).

Provisioning requires that a provisioner is included in the image. There are two provisioners:

- [Azure Linux Agent](#)
- [cloud-init](#)

These are [prerequisites](#) for creating an image.

### Specialized images

These are images that are completely configured and do not require VM or special parameters. The platform will just turn the VM on and you will need to handle uniqueness within the VM, like setting a hostname, to avoid DNS conflicts on the same VNET.

Provisioning agents are not required for these images, however you may want to have extension handling capabilities. You can install the Linux Agent but disable the provisioning option. Even though you do not need a provisioning agent, the image must fulfill [prerequisites](#) for Azure Images.

## Image storage options

When bringing your Linux image you have two options:

- Managed images for simple VM creation in a development and test environment.
- [Azure Compute Gallery](#) for creating and sharing images at-scale.

### Managed images

Managed images can be used to create multiple VMs, but they have a lot of limitations. Managed images can only be created from a generalized source (VM or VHD). They can only be used to create VMs in the same region and they can't be shared across subscriptions and tenants.

Managed images can be used for development and test environments, where you need a couple of simple generalized images to use within single region and subscription.

### Azure Compute Gallery

[Azure Compute Gallery](#) (formerly known as Shared Image Gallery) is recommended for creating, managing, and sharing images at scale. Azure Compute Gallery helps you build structure and organization around your images.

- Support for both generalized and specialized images.
- Support for image both generation 1 and 2 images.
- Global replication of images.
- Versioning and grouping of images for easier management.
- Highly available images with Zone Redundant Storage (ZRS) in regions that support Availability Zones. ZRS offers better resilience against zonal failures.
- Sharing across subscriptions and even between Active Directory (AD) tenants using Azure RBAC.
- Scaling your deployments with image replicas in each region.

At a high level, you create a gallery and it is made up of:

- Image Definitions - These are containers that hold groups of images.
- Image Versions - These are the actual images.

## Hyper-V generation

Azure supports Hyper-V Generation 1 (Gen1) and Generation 2 (Gen2). Gen2 is the latest generation and offers additional functionality over Gen1. For example: increased memory, Intel Software Guard Extensions (Intel SGX), and virtualized persistent memory (vPMEM). Generation 2 VMs running on-premises have some features that aren't supported in Azure yet. For more information, see the Features and capabilities section in this [article](#). Create Gen2 images if you require the additional functionality.

If you still need to create your own image, ensure it meets the [image prerequisites](#) and upload to Azure.

Distribution specific requirements:

- [CentOS-based Distributions](#)
- [Debian Linux](#)
- [Flatcar Container Linux](#)
- [Oracle Linux](#)
- [Red Hat Enterprise Linux](#)
- [SLES & openSUSE](#)
- [Ubuntu](#)

## Next steps

Learn how to create an [Azure Compute Gallery](#).

# Azure Linux VM provisioning

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you create a VM from a generalized image (Azure Compute Gallery or Managed Image), the control plane will allow you to create a VM, and pass parameters and settings to the VM. This is called *VM provisioning*.

During provisioning, the platform makes required VM Create parameter values (hostname, username, password, SSH keys, customData) available to the VM as it boots.

A provisioning agent baked inside the image will interface with the platform, connecting up to multiple independent provisioning interfaces), set the properties and signal to the platform it has completed.

The provisioning agents can be the [Azure Linux Agent](#), or [cloud-init](#). These are [prerequisites](#) of creating generalized images.

The provisioning agents, provide support for all endorsed [Azure Linux distributions](#), and you will find the endorsed distro images in many cases will ship with both cloud-init and the Linux Agent. This gives you the option to have cloud-init to handle the provisioning, then the Linux Agent will provide support to handle [Azure Extensions](#). Providing support for extensions means the VM will then be eligible to support additional Azure services, such as VM Password Reset, Azure Monitoring, Azure Backup, Azure Disk encryption etc.

After provisioning completes, cloud-init will run on each boot. Cloud-init will monitor for changes to the VM, like networking changes, mounting, and formatting the ephemeral disk, and starting the Linux Agent. The Linux Agent continually runs on the server, seeking a 'goal state' (new configuration) from the Azure platform, so whenever you install extensions, the agent will be able to process them.

Whilst there are currently two provisioning agents, cloud-init should be the provisioning agent you choose, and the Linux Agent should be installed for extension support. This allows you to take advantage of platform optimizations, and allows you to disable/remove the Linux Agent, for more details on how to create images without the agent, and how to remove it, please review this [documentation](#).

If you have a Linux kernel that cannot support running either agent, but wish to set some of the VM Create properties, such as hostname, customData, userName, password, ssh keys, then in this document discusses how you can [create generalized images without an agent](#), and meet platform requirements.

## Provisioning agent responsibilities

### Image provisioning

- Creation of a user account
- Configuring SSH authentication types
- Deployment of SSH public keys and key pairs
- Setting the host name
- Publishing the host name to the platform DNS
- Reporting SSH host key fingerprint to the platform
- Resource Disk Management
- Formatting and mounting the resource disk
- Consuming and processing `customData`

### Networking

- Manages routes to improve compatibility with platform DHCP servers
- Ensures the stability of the network interface name

## Kernel

- Configures virtual NUMA (disable for kernel < 2.6.37 )
- Consumes Hyper-V entropy for /dev/random
- Configures SCSI timeouts for the root device (which could be remote)

## Diagnostics

- Console redirection to the serial port

# Communication

The information flow from the platform to the agent occurs via two channels:

- A boot-time attached DVD for IaaS deployments. The DVD includes an OVF-compliant configuration file that includes all provisioning information, other than the actual SSH key pairs.
- A TCP endpoint exposing a REST API used to obtain deployment, and topology configuration.

# Azure provisioning agent requirements

The Linux Agent, and cloud-init, depend on some system packages in order to function properly:

- Python 2.6+
- OpenSSL 1.0+
- OpenSSH 5.3+
- Filesystem utilities: sfdisk , fdisk , mkfs , parted
- Password tools: chpasswd, sudo
- Text processing tools: sed, grep
- Network tools: ip-route
- Kernel support for mounting UDF filesystems.

# Next steps

If you need to, you can [disable provisioning and remove the Linux agent](#).

# Endorsed Linux distributions on Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

Partners provide Linux images in Azure Marketplace. Microsoft works with various Linux communities to add even more flavors to the Endorsed Distribution list. For distributions that are not available from the Marketplace, you can always bring your own Linux by following the guidelines at [Create and upload a virtual hard disk that contains the Linux operating system](#).

## Supported distributions and versions

The following table lists the Linux distributions and versions that are supported on Azure. For more information, see [Support for Linux images in Microsoft Azure](#).

The Linux Integration Services (LIS) drivers for Hyper-V and Azure are kernel modules that Microsoft contributes directly to the upstream Linux kernel. Some LIS drivers are built into the distribution's kernel by default. Older distributions that are based on Red Hat Enterprise (RHEL)/CentOS are available as a separate download at [Linux Integration Services Version 4.2 for Hyper-V and Azure](#). For more information, see [Linux kernel requirements](#).

The Azure Linux Agent is already pre-installed on Azure Marketplace images and is typically available from the distribution's package repository. Source code can be found on [GitHub](#).

DISTRIBUTION	VERSION	DRIVERS	AGENT
CentOS by Rogue Wave Software (formerly known as OpenLogic)	CentOS 6.x, 7.x, 8.x	CentOS 6.3: <a href="#">LIS download</a> CentOS 6.4+: In kernel	Package: In <a href="#">repo</a> under "WALinuxAgent" Source code: <a href="#">GitHub</a>
<a href="#">CoreOS</a> CoreOS is now <a href="#">end of life</a> as of May 26, 2020.	No Longer Available		
Debian by credativ	9.x (LTS), 10.x, 11.x	In kernel	Package: In repo under "waagent" Source code: <a href="#">GitHub</a>
Flatcar Container Linux by Kinvolk	Pro, Stable, Beta	In kernel	wa-linux-agent is installed already in /usr/share/oem/bin/waagent
Oracle Linux by Oracle	6.x, 7.x, 8.x	In kernel	Package: In repo under "WALinuxAgent" Source code: <a href="#">GitHub</a>
<a href="#">Red Hat Enterprise Linux by Red Hat</a>	7.x, 8.x	In kernel	Package: In repo under "WALinuxAgent" Source code: <a href="#">GitHub</a>

DISTRIBUTION	VERSION	DRIVERS	AGENT
SUSE Linux Enterprise by SUSE	SLES/SLES for SAP 11.x, 12.x, 15.x <a href="#">SUSE Public Cloud Image Lifecycle</a>	In kernel	Package: for 11 in <a href="#">Cloud:Tools</a> repo for 12 included in "Public Cloud" Module under "python-azure-agent" Source code: <a href="#">GitHub</a>
openSUSE by SUSE	openSUSE Leap 15.x	In kernel	Package: In <a href="#">Cloud:Tools</a> repo under "python-azure-agent" Source code: <a href="#">GitHub</a>
Ubuntu by Canonical	Ubuntu Server and Pro. 18.x, 20.x 22.x Information about extended support for Ubuntu 14.04 pro and 16.04 pro can be found here: <a href="#">Ubuntu Extended Security Maintenance</a> .	In kernel	Package: In repo under "walinuxagent" Source code: <a href="#">GitHub</a>

## Image update cadence

Azure requires that the publishers of the endorsed Linux distributions regularly update their images in Azure Marketplace with the latest patches and security fixes, at a quarterly or faster cadence. Updated images in the Marketplace are available automatically to customers as new versions of an image SKU. More information about how to find Linux images: [Find Linux VM images in Azure Marketplace](#).

## Azure-tuned kernels

Azure works closely with various endorsed Linux distributions to optimize the images that they published to Azure Marketplace. One aspect of this collaboration is the development of "tuned" Linux kernels that are optimized for the Azure platform and delivered as fully supported components of the Linux distribution. The Azure-Tuned kernels incorporate new features and performance improvements, and at a faster (typically quarterly) cadence compared to the default or generic kernels that are available from the distribution.

In most cases, you will find these kernels pre-installed on the default images in Azure Marketplace so customers will immediately get the benefit of these optimized kernels. More information about these Azure-Tuned kernels can be found in the following links:

- [CentOS Azure-Tuned Kernel - Available via the CentOS Virtualization SIG](#)
- [Debian Cloud Kernel - Available with the Debian 10 and Debian 9 "backports" image on Azure](#)
- [SLES Azure-Tuned Kernel](#)
- [Ubuntu Azure-Tuned Kernel](#)
- [Flatcar Container Linux Pro](#)

## Partners

### CoreOS

CoreOS is scheduled to be [end of life](#) by May 26, 2020. Microsoft has two (2) channels of migration for CoreOS

users.

- Flatcar by Kinvolk (see the "Flatcar Container Linux by Kinvolk" entry.)
- [Fedora Core OS](#) (customers must upload their own image. Here is the [migration documentation](#)).

## **credativ**

<https://www.creativ.de/en/portfolio/support/open-source-support-center/>

credativ is an independent consulting and services company that specializes in the development and implementation of professional solutions by using free software. As leading open-source specialists, creditativ has international recognition with many IT departments that use their support. In conjunction with Microsoft, creditativ is preparing Debian images. The images are specially designed to run on Azure and can be easily managed via the platform. creditativ will also support the long-term maintenance and updating of the Debian images for Azure through its Open Source Support Centers.

## **Kinvolk**

<https://www.flatcar-linux.org/>

Kinvolk is the team behind Flatcar Container Linux, continuing the original CoreOS vision for a minimal, immutable, and auto-updating foundation for containerized applications. As a minimal distro, Flatcar contains just those packages required for deploying containers. Its immutable file system guarantees consistency and security, while its auto-update capabilities, enable you to be always up-to-date with the latest security fixes. Kinvolk was [acquired by Microsoft](#) in April 2021 and, post-acquisition, continues its mission to support the Flatcar Container Linux community.

## **Oracle**

<https://www.oracle.com/technetwork/topics/cloud/faq-1963009.html>

Oracle's strategy is to offer a broad portfolio of solutions for public and private clouds. The strategy gives customers choice and flexibility in how they deploy Oracle software in Oracle clouds and other clouds. Oracle's partnership with Microsoft enables customers to deploy Oracle software in Microsoft public and private clouds with the confidence of certification and support from Oracle. Oracle's commitment and investment in Oracle public and private cloud solutions is unchanged.

## **Red Hat**

<https://www.redhat.com/en/partners/strategic-alliance/microsoft>

The world's leading provider of open-source solutions, Red Hat helps more than 90% of Fortune 500 companies solve business challenges, align their IT and business strategies, and prepare for the future of technology. Red Hat achieves this by providing secure solutions through an open business model and an affordable, predictable subscription model.

## **SUSE**

<https://www.suse.com/suse-linux-enterprise-server-on-azure>

SUSE Linux Enterprise Server on Azure is a proven platform that provides superior reliability and security for cloud computing. SUSE's versatile Linux platform seamlessly integrates with Azure cloud services to deliver an easily manageable cloud environment. With more than 9,200 certified applications from more than 1,800 independent software vendors for SUSE Linux Enterprise Server, SUSE ensures that workloads running supported in the data center can be confidently deployed on Azure.

## **Canonical**

<https://www.ubuntu.com/cloud/azure>

Canonical engineering and open community governance drive Ubuntu's success in client, server, and cloud computing, which includes personal cloud services for consumers. Canonical's vision of a unified, free platform

in Ubuntu, from phone to cloud, provides a family of coherent interfaces for the phone, tablet, TV, and desktop. This vision makes Ubuntu the first choice for diverse institutions from public cloud providers to the makers of consumer electronics and a favorite among individual technologists.

With developers and engineering centers around the world, Canonical is uniquely positioned to partner with hardware makers, content providers, and software developers to bring Ubuntu solutions to market for PCs, servers, and handheld devices.

# Information for community supported and non-endorsed distributions

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The Azure platform SLA applies to virtual machines running the Linux OS only when one of the [endorsed distributions](#) is used. For these endorsed distributions, pre-configured Linux images are provided in the Azure Marketplace.

- [Linux on Azure - Endorsed Distributions](#)
- [Support for Linux images in Microsoft Azure](#)

All other, non-Azure Marketplace, distributions running on Azure have a number of prerequisites. This article can't be comprehensive, as every distribution is different. Even if you meet all the criteria below, you may need to significantly tweak your Linux system for it to run properly.

This article focuses on general guidance for running your Linux distribution on Azure.

## General Linux Installation Notes

1. The Hyper-V virtual hard disk (VHDX) format isn't supported in Azure, only *fixed VHD*. You can convert the disk to VHD format using Hyper-V Manager or the [Convert-VHD](#) cmdlet. If you're using VirtualBox, select **Fixed size** rather than the default (dynamically allocated) when creating the disk.
2. Azure supports Gen1 (BIOS boot) & Gen2 (UEFI boot) Virtual machines.
3. The maximum size allowed for the VHD is 1,023 GB.
4. When installing the Linux system we recommend that you use standard partitions, rather than Logical Volume Manager (LVM) which is the default for many installations. Using standard partitions will avoid LVM name conflicts with cloned VMs, particularly if an OS disk is ever attached to another identical VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks.
5. Kernel support for mounting UDF file systems is necessary. At first boot on Azure the provisioning configuration is passed to the Linux VM by using UDF-formatted media that is attached to the guest. The Azure Linux agent must mount the UDF file system to read its configuration and provision the VM.
6. Linux kernel versions earlier than 2.6.37 don't support NUMA on Hyper-V with larger VM sizes. This issue primarily impacts older distributions using the upstream Red Hat 2.6.32 kernel, and was fixed in Red Hat Enterprise Linux (RHEL) 6.6 (kernel-2.6.32-504). Systems running custom kernels older than 2.6.37, or RHEL-based kernels older than 2.6.32-504 must set the boot parameter `numa=off` on the kernel command line in grub.conf. For more information, see [Red Hat KB 436883](#).
7. Don't configure a swap partition on the OS disk. The Linux agent can be configured to create a swap file on the temporary resource disk, as described in the following steps.
8. All VHDs on Azure must have a virtual size aligned to 1 MB ( $1024 \times 1024$  bytes). When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1 MB before conversion, as described in the following steps.
9. Use the most up-to-date distribution version, packages, and software.

10. Remove users and system accounts, public keys, sensitive data, unnecessary software and application.

## Installing kernel modules without Hyper-V

Azure runs on the Hyper-V hypervisor, so Linux requires certain kernel modules to run in Azure. If you have a VM that was created outside of Hyper-V, the Linux installers may not include the drivers for Hyper-V in the initial ramdisk (initrd or initramfs), unless the VM detects that it's running on a Hyper-V environment. When using a different virtualization system (such as VirtualBox, KVM, and so on) to prepare your Linux image, you may need to rebuild the initrd so that at least the hv\_vmbus and hv\_storvsc kernel modules are available on the initial ramdisk. This known issue is for systems based on the upstream Red Hat distribution, and possibly others.

The mechanism for rebuilding the initrd or initramfs image may vary depending on the distribution. Consult your distribution's documentation or support for the proper procedure. Here is one example for rebuilding the initrd by using the `mkinitrd` utility:

1. Back up the existing initrd image:

```
cd /boot  
sudo cp initrd-`uname -r`.img initrd-`uname -r`.img.bak
```

2. Rebuild the `initrd` with the `hv_vmbus` and `hv_storvsc` kernel modules:

```
sudo mkinitrd --preload=hv_storvsc --preload=hv_vmbus -v -f initrd-`uname -r`.img `uname -r`
```

## Resizing VHDs

VHD images on Azure must have a virtual size aligned to 1 MB. Typically, VHDs created using Hyper-V are aligned correctly. If the VHD isn't aligned correctly, you may receive an error message similar to the following when you try to create an image from your VHD.

```
The VHD http://<mystorageaccount>.blob.core.windows.net/vhds/MyLinuxVM.vhd has an unsupported virtual size of 21475270656 bytes. The size must be a whole number (in MBs).
```

In this case, resize the VM using either the Hyper-V Manager console or the [Resize-VHD](#) PowerShell cmdlet. If you aren't running in a Windows environment, we recommend using `qemu-img` to convert (if needed) and resize the VHD.

### NOTE

There is a [known bug in qemu-img](#) versions >=2.2.1 that results in an improperly formatted VHD. The issue has been fixed in QEMU 2.6. We recommend using either `qemu-img` 2.2.0 or lower, or 2.6 or higher.

1. Resizing the VHD directly using tools such as `qemu-img` or `vbox-manage` may result in an unbootable VHD. We recommend first converting the VHD to a RAW disk image. If the VM image was created as a RAW disk image (the default for some hypervisors such as KVM), then you may skip this step.

```
qemu-img convert -f vpc -O raw MyLinuxVM.vhd MyLinuxVM.raw
```

2. Calculate the required size of the disk image so that the virtual size is aligned to 1 MB. The following bash shell script uses `qemu-img info` to determine the virtual size of the disk image, and then calculates the size to the next 1 MB.

```

rawdisk="MyLinuxVM.raw"
vhddisk="MyLinuxVM.vhd"

MB=$((1024*1024))
size=$(qemu-img info -f raw --output json "$rawdisk" | \
gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')

rounded_size=$((($size+$MB-1)/$MB)*$MB)

echo "Rounded Size = $rounded_size"

```

3. Resize the raw disk using `$rounded_size` as set above.

```
qemu-img resize MyLinuxVM.raw $rounded_size
```

4. Now, convert the RAW disk back to a fixed-size VHD.

```
qemu-img convert -f raw -o subformat=fixed -O vpc MyLinuxVM.raw MyLinuxVM.vhd
```

Or, with qemu version 2.6+, include the `force_size` option.

```
qemu-img convert -f raw -o subformat=fixed,force_size -O vpc MyLinuxVM.raw MyLinuxVM.vhd
```

## Linux Kernel Requirements

The Linux Integration Services (LIS) drivers for Hyper-V and Azure are contributed directly to the upstream Linux kernel. Many distributions that include a recent Linux kernel version (such as 3.x) have these drivers available already, or otherwise provide backported versions of these drivers with their kernels. These drivers are constantly being updated in the upstream kernel with new fixes and features, so when possible we recommend running an [endorsed distribution](#) that includes these fixes and updates.

If you're running a variant of Red Hat Enterprise Linux versions 6.0 to 6.3, then you'll need to install the [latest LIS drivers for Hyper-V](#). Beginning with RHEL 6.4+ (and derivatives) the LIS drivers are already included with the kernel and so no additional installation packages are needed.

If a custom kernel is required, we recommend a recent kernel version (such as 3.8+). For distributions or vendors who maintain their own kernel, you'll need to regularly backport the LIS drivers from the upstream kernel to your custom kernel. Even if you're already running a relatively recent kernel version, we highly recommend keeping track of any upstream fixes in the LIS drivers and backport them as needed. The locations of the LIS driver source files are specified in the [MAINTAINERS](#) file in the Linux kernel source tree:

```

F:    arch/x86/include/asm/mshyperv.h
F:    arch/x86/include/uapi/asm/hyperv.h
F:    arch/x86/kernel/cpu/mshyperv.c
F:    drivers/hid/hid-hyperv.c
F:    drivers/hv/
F:    drivers/input/serio/hyperv-keyboard.c
F:    drivers/net/hyperv/
F:    drivers/scsi/storvsc_drv.c
F:    drivers/video/fbdev/hyperv_fb.c
F:    include/linux/hyperv.h
F:    tools/hv/

```

The following patches must be included in the kernel. This list can't be complete for all distributions.

- [ata\\_piix](#): defer disks to the Hyper-V drivers by default
- [storvsc](#): Account for in-transit packets in the RESET path
- [storvsc](#): avoid usage of WRITE\_SAME
- [storvsc](#): Disable WRITE SAME for RAID and virtual host adapter drivers
- [storvsc](#): NULL pointer dereference fix
- [storvsc](#): ring buffer failures may result in I/O freeze
- [scsi\\_sysfs](#): protect against double execution of \_\_scsi\_remove\_device

## The Azure Linux Agent

The [Azure Linux Agent](#) `waagent` provisions a Linux virtual machine in Azure. You can get the latest version, file issues, or submit pull requests at the [Linux Agent GitHub repo](#).

- The Linux agent is released under the Apache 2.0 license. Many distributions already provide RPM or .deb packages for the agent, and these packages can easily be installed and updated.
- The Azure Linux Agent requires Python v2.6+.
- The agent also requires the `python-pyasn1` module. Most distributions provide this module as a separate package to be installed.
- In some cases, the Azure Linux Agent may not be compatible with NetworkManager. Many of the RPM/deb packages provided by distributions configure NetworkManager as a conflict to the `waagent` package. In these cases, it will uninstall NetworkManager when you install the Linux agent package.
- The Azure Linux Agent must be at or above the [minimum supported version](#).

### NOTE

Make sure '`udf`' (cloud-init >= 21.2) and '`vfat`' modules are enable. Blocklisting the `udf` module will cause a provisioning failure and backlisting `vfat` module will cause both provisioning and boot failures. *Cloud-init < 21.2 are not affected and does not require this change.*

## General Linux System Requirements

1. Modify the kernel boot line in GRUB or GRUB2 to include the following parameters, so that all console messages are sent to the first serial port. These messages can assist Azure support with debugging any issues.

```
GRUB_CMDLINE_LINUX="rootdelay=300 console=ttyS0 earlyprintk=ttyS0 net.ifnames=0"
```

We also recommend *removing* the following parameters if they exist.

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot isn't useful in a cloud environment, where we want all logs sent to the serial port. The `crashkernel` option may be left configured if needed, but note that this parameter reduces the amount of available memory in the VM by at least 128 MB, which may be problematic for smaller VM sizes.

2. After you are done editing `/etc/default/grub`, run the following command to rebuild the grub configuration:

```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

3. Add Hyper-V modules both initrd and initramfs instructions (Dracut).

4. Rebuild initrd or initramfs **Initramfs**

```
cp /boot/initramfs-$(uname -r).img /boot/initramfs-[latest kernel version ].img.bak  
dracut -f -v /boot/initramfs-[latest kernel version ].img [depending on the version of grub]  
grub-mkconfig -o /boot/grub/grub.cfg  
grub2-mkconfig -o /boot/grub2/grub.cfg
```

## Initrd

```
mv /boot/[initrd kernel] /boot/[initrd kernel]-old  
mkinitrd /boot/initrd.img-[initrd kernel]-generic /boot/[initrd kernel]-generic-old  
update-initramfs -c -k [initrd kernel]  
update-grub
```

5. Ensure that the SSH server is installed, and configured to start at boot time. This configuration is usually the default.

6. Install the Azure Linux Agent. The Azure Linux Agent is required for provisioning a Linux image on Azure. Many distributions provide the agent as an RPM or .deb package (the package is typically called WALinuxAgent or walinuagent). The agent can also be installed manually by following the steps in the [Linux Agent Guide](#).

Install the Azure Linux Agent, cloud-init and other necessary utilities by running the following command:

### Redhat/Centos

```
sudo yum install -y [waagent] cloud-init cloud-utils-growpart gdisk hyperv-daemons
```

### Ubuntu/Debian

```
sudo apt install walinuagent cloud-init cloud-utils-growpart gdisk hyperv-daemons
```

### Suse

```
sudo zypper install python-azure-agent cloud-init cloud-utils-growpart gdisk hyperv-daemons
```

Then enable the agent and cloud-init on all distributions using:

```
sudo systemctl enable waagent.service  
sudo systemctl enable cloud-init.service
```

7. Don't create swap space on the OS disk. The Azure Linux Agent can automatically configure swap space using the local resource disk that is attached to the VM after provisioning on Azure. The local resource disk is a temporary disk, and might be emptied when the VM is deprovisioned. After installing the Azure Linux Agent, modify the following parameters in /etc/waagent.conf as needed.

```
ResourceDisk.Format=  
ResourceDisk.Filesystem=ext4  
ResourceDisk.MountPoint=/mnt/resource  
ResourceDisk.EnableSwap=  
ResourceDisk.SwapSizeMB=2048 ## NOTE: Set this to your desired size.
```

8. Configure cloud-init to handle the provisioning:

a. Configure waagent for cloud-init:

```
sed -i 's/Provisioning.Agent=auto/Provisioning.Agent=cloud-init/g' /etc/waagent.conf
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf
```

If you are migrating a specific virtual machine and do not wish to create a generalized image, set

`Provisioning.Agent=disabled` in the `/etc/waagent.conf` config.

b. Configure mounts:

```
echo "Adding mounts and disk_setup to init stage"
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg
```

c. Configure Azure datasource:

```
echo "Allow only Azure datasource, disable fetching network setting via IMDS"
cat > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg <<EOF
datasource_list: [ Azure ]
datasource:
  Azure:
    apply_network_config: False
EOF
```

d. If configured, remove existing swapfile:

```
if [[ -f /mnt/resource/swapfile ]]; then
echo "Removing swapfile" #RHEL uses a swapfile by default
swapoff /mnt/resource/swapfile
rm /mnt/resource/swapfile -f
fi
```

9. Configure cloud-init logging:

```
echo "Add console log file"
cat >> /etc/cloud/cloud.cfg.d/05_logging.cfg <<EOF

# This tells cloud-init to redirect its stdout and stderr to
# 'tee -a /var/log/cloud-init-output.log' so the user can see output
# there without needing to look on the console.
output: {all: '| tee -a /var/log/cloud-init-output.log'}
EOF
```

10. Swap configuration. Do not create swap space on the operating system disk. Previously, the Azure Linux Agent automatically configured swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However, this is now handled by cloud-init, you must not use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=n
ResourceDisk.EnableSwap=n
```

If you want to mount, format and create swap you can either: 1. Pass this in as a cloud-init config every time you create a VM through `customdata`. This is the recommended method. 2. Use a cloud-init directive baked into the image that will do this every time the VM is created.

```
echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
ephemeral0:
  table_type: mbr
  layout: [66, [33, 82]]
  overwrite: True
fs_setup:
- device: ephemeral0.1
  filesystem: ext4
- device: ephemeral0.2
  filesystem: swap
mounts:
- ["ephemeral0.1", "/mnt"]
- ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service,x-
systemd.device-timeout=2", "0", "0"]
EOF
```

```

## 11. Deprovision.

### Caution

If you are migrating a specific virtual machine and do not wish to create a generalized image, skip the deprovision step. Running the command `waagent -force -deprovision+user` will render the source machine unusable, this step is intended only to create a generalized image.

Run the following commands to deprovision the virtual machine.

```
# sudo rm -f /var/log/waagent.log
# sudo cloud-init clean
# waagent -force -deprovision+user
# rm -f ~/.bash_history
# export HISTSIZE=0
# logout
```

### NOTE

On Virtualbox you may see the following error after running `waagent -force -deprovision` that says `[Errno 5] Input/output error`. This error message is not critical and can be ignored.

## 12. Shut down the virtual machine and upload the VHD to Azure.

# Next Steps

[Create a Linux VM from a custom disk with the Azure CLI.](#)

# Prepare a CentOS-based virtual machine for Azure

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Learn to create and upload an Azure virtual hard disk (VHD) that contains a CentOS-based Linux operating system.

- [Prepare a CentOS 6.x virtual machine for Azure](#)
- [Prepare a CentOS 7.0+ virtual machine for Azure](#)

## Prerequisites

This article assumes that you have already installed a CentOS (or similar derivative) Linux operating system to a virtual hard disk. Multiple tools exist to create .vhd files, for example a virtualization solution such as Hyper-V. For instructions, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

### CentOS installation notes

- Please see also [General Linux Installation Notes](#) for more tips on preparing Linux for Azure.
- The VHDX format is not supported in Azure, only **fixed VHD**. You can convert the disk to VHD format using Hyper-V Manager or the convert-vhd cmdlet. If you are using VirtualBox this means selecting **Fixed size** as opposed to the default dynamically allocated when creating the disk.
- When installing the Linux system it is *recommended* that you use standard partitions rather than LVM (often the default for many installations). This will avoid LVM name conflicts with cloned VMs, particularly if an OS disk ever needs to be attached to another identical VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks.
- **Kernel support for mounting UDF file systems is required.** At first boot on Azure the provisioning configuration is passed to the Linux VM via UDF-formatted media that is attached to the guest. The Azure Linux agent must be able to mount the UDF file system to read its configuration and provision the VM.
- Linux kernel versions below 2.6.37 do not support NUMA on Hyper-V with larger VM sizes. This issue primarily impacts older distributions using the upstream Red Hat 2.6.32 kernel, and was fixed in RHEL 6.6 (kernel-2.6.32-504). Systems running custom kernels older than 2.6.37, or RHEL-based kernels older than 2.6.32-504 must set the boot parameter `numa=off` on the kernel command-line in grub.conf. For more information see Red Hat [KB 436883](#).
- Do not configure a swap partition on the OS disk. More information about this can be found in the steps below.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1MB before conversion. See [Linux Installation Notes](#) for more information.

## CentOS 6.x

1. In Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open a console window for the virtual machine.
3. In CentOS 6, NetworkManager can interfere with the Azure Linux agent. Uninstall this package by running the following command:

```
sudo rpm -e --nodeps NetworkManager
```

4. Create or edit the file `/etc/sysconfig/network` and add the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

5. Create or edit the file `/etc/sysconfig/network-scripts/ifcfg-eth0` and add the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no
```

6. Modify udev rules to avoid generating static rules for the Ethernet interface(s). These rules can cause problems when cloning a virtual machine in Microsoft Azure or Hyper-V:

```
sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules  
sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

7. Ensure the network service will start at boot time by running the following command:

```
sudo chkconfig network on
```

8. If you would like to use the OpenLogic mirrors that are hosted within the Azure datacenters, then replace the `/etc/yum.repos.d/CentOS-Base.repo` file with the following repositories. This will also add the `[openlogic]` repository that includes additional packages such as the Azure Linux agent:

```

[openlogic]
name=CentOS-$releasever - openlogic packages for $basearch
baseurl=http://olcentgbl.trafficmanager.net/openlogic/$releasever/openlogic/$basearch/
enabled=1
gpgcheck=0

[base]
name=CentOS-$releasever - Base
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6

#released updates
[updates]
name=CentOS-$releasever - Updates
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6

#additional packages that may be useful
[extras]
name=CentOS-$releasever - Extras
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=extras&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/extras/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6

#additional packages that extend functionality of existing packages
[centosplus]
name=CentOS-$releasever - Plus
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=centosplus&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/centosplus/$basearch/
gpgcheck=1
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6

#contrib - packages by Centos Users
[contrib]
name=CentOS-$releasever - Contrib
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=contrib&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/contrib/$basearch/
gpgcheck=1
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-6

```

#### NOTE

The rest of this guide will assume you are using at least the `[openlogic]` repo, which will be used to install the Azure Linux agent below.

- Add the following line to `/etc/yum.conf`:

```
http_caching=packages
```

- Run the following command to clear the current yum metadata and update the system with the latest packages:

```
yum clean all
```

Unless you are creating an image for an older version of CentOS, it is recommended to update all the packages to the latest:

```
sudo yum -y update
```

A reboot may be required after running this command.

11. (Optional) Install the drivers for the Linux Integration Services (LIS).

**IMPORTANT**

The step is **required** for CentOS 6.3 and earlier, and optional for later releases.

```
sudo rpm -e hypervkvpd ## (may return error if not installed, that's OK)
sudo yum install microsoft-hyper-v
```

Alternatively, you can follow the manual installation instructions on the [LIS download page](#) to install the RPM onto your VM.

12. Install the Azure Linux Agent and dependencies. Start and enable waagent service:

```
sudo yum install python-pyasn1 WALinuxAgent
sudo service waagent start
sudo chkconfig waagent on
```

The WALinuxAgent package will remove the NetworkManager and NetworkManager-gnome packages if they were not already removed as described in step 3.

13. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure.

To do this, open `/boot/grub/menu.lst` in a text editor and ensure that the default kernel includes the following parameters:

```
console=ttyS0 earlyprintk=ttyS0 rootdelay=300
```

This will also ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues.

In addition to the above, it is recommended to *remove* the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. The `crashkernel` option may be left configured if desired, but note that this parameter will reduce the amount of available memory in the VM by 128MB or more, which may be problematic on the smaller VM sizes.

**IMPORTANT**

CentOS 6.5 and earlier must also set the kernel parameter `numa=off`. See Red Hat [KB 436883](#).

14. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.

15. Do not create swap space on the OS disk.

The Azure Linux Agent can automatically configure swap space using the local resource disk that is attached to the VM after provisioning on Azure. Note that the local resource disk is a *temporary* disk, and might be emptied when the VM is deprovisioned. After installing the Azure Linux Agent (see previous step), modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=y
ResourceDisk.Filesystem=ext4
ResourceDisk.MountPoint=/mnt/resource
ResourceDisk.EnableSwap=y
ResourceDisk.SwapSizeMB=2048 ## NOTE: set this to whatever you need it to be.
```

16. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
sudo waagent -force -deprovision+user
export HISTSIZE=0
logout
```

17. Click **Action** -> **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## CentOS 7.0+

### Changes in CentOS 7 (and similar derivatives)

Preparing a CentOS 7 virtual machine for Azure is very similar to CentOS 6, however there are several important differences worth noting:

- The NetworkManager package no longer conflicts with the Azure Linux agent. This package is installed by default and we recommend that it is not removed.
- GRUB2 is now used as the default bootloader, so the procedure for editing kernel parameters has changed (see below).
- XFS is now the default file system. The ext4 file system can still be used if desired.
- Since CentOS 8 Stream and newer no longer include `network.service` by default, you will need to install it manually:

```
sudo yum install network-scripts
sudo systemctl enable network.service
```

### Configuration Steps

1. In Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open a console window for the virtual machine.
3. Create or edit the file `/etc/sysconfig/network` and add the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

4. Create or edit the file `/etc/sysconfig/network-scripts/ifcfg-eth0` and add the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no  
NM_CONTROLLED=no
```

5. Modify udev rules to avoid generating static rules for the Ethernet interface(s). These rules can cause problems when cloning a virtual machine in Microsoft Azure or Hyper-V:

```
sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules
```

6. If you would like to use the OpenLogic mirrors that are hosted within the Azure datacenters, then replace the `/etc/yum.repos.d/CentOS-Base.repo` file with the following repositories. This will also add the `[openlogic]` repository that includes packages for the Azure Linux agent:

```

[openlogic]
name=CentOS-$releasever - openlogic packages for $basearch
baseurl=http://olcentgbl.trafficmanager.net/openlogic/$releasever/openlogic/$basearch/
enabled=1
gpgcheck=0

[base]
name=CentOS-$releasever - Base
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7

#released updates
[updates]
name=CentOS-$releasever - Updates
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7

#additional packages that may be useful
[extras]
name=CentOS-$releasever - Extras
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=extras&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/extras/$basearch/
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7

#additional packages that extend functionality of existing packages
[centosplus]
name=CentOS-$releasever - Plus
#mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=centosplus&infra=$infra
baseurl=http://olcentgbl.trafficmanager.net/centos/$releasever/centosplus/$basearch/
gpgcheck=1
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7

```

#### NOTE

The rest of this guide will assume you are using at least the `[openlogic]` repo, which will be used to install the Azure Linux agent below.

- Run the following command to clear the current yum metadata and install any updates:

```
sudo yum clean all
```

Unless you are creating an image for an older version of CentOS, it is recommended to update all the packages to the latest:

```
sudo yum -y update
```

A reboot maybe required after running this command.

- Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this, open `/etc/default/grub` in a text editor and edit the `GRUB_CMDLINE_LINUX` parameter, for example:

```
GRUB_CMDLINE_LINUX="rootdelay=300 console=ttyS0 earlyprintk=ttyS0 net.ifnames=0"
```

This will also ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues. It also turns off the new CentOS 7 naming conventions for NICs. In addition to the above, it is recommended to *remove* the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. The `crashkernel` option may be left configured if desired, but note that this parameter will reduce the amount of available memory in the VM by 128MB or more, which may be problematic on the smaller VM sizes.

- Once you are done editing `/etc/default/grub` per above, run the following command to rebuild the grub configuration:

```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

- If building the image from **VMware**, **VirtualBox** or **KVM**: Ensure the Hyper-V drivers are included in the initramfs:

Edit `/etc/dracut.conf`, add content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild the initramfs:

```
sudo dracut -f -v
```

- Install the Azure Linux Agent and dependencies for Azure VM Extensions:

```
sudo yum install python-pyasn1 WALinuxAgent
sudo systemctl enable waagent
```

- Install cloud-init to handle the provisioning

```

yum install -y cloud-init cloud-utils-growpart gdisk hyperv-daemons

# Configure waagent for cloud-init
sed -i 's/Provisioning.UseCloudInit=n/Provisioning.UseCloudInit=y/g' /etc/waagent.conf
sed -i 's/Provisioning.Enabled=y/Provisioning.Enabled=n/g' /etc/waagent.conf

echo "Adding mounts and disk_setup to init stage"
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg

echo "Allow only Azure datasource, disable fetching network setting via IMDS"
cat > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg <<EOF
datasource_list: [ Azure ]
datasource:
  Azure:
    apply_network_config: False
EOF

if [[ -f /mnt/resource/swapfile ]]; then
echo Removing swapfile - RHEL uses a swapfile by default
swapoff /mnt/resource/swapfile
rm /mnt/resource/swapfile -f
fi

echo "Add console log file"
cat >> /etc/cloud/cloud.cfg.d/05_logging.cfg <<EOF

# This tells cloud-init to redirect its stdout and stderr to
# 'tee -a /var/log/cloud-init-output.log' so the user can see output
# there without needing to look on the console.
output: {all: '| tee -a /var/log/cloud-init-output.log'}
EOF

```

### 13. Swap configuration Do not create swap space on the operating system disk.

Previously, the Azure Linux Agent was used automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However this is now handled by cloud-init you **must not** use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```

sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf

```

If you want mount, format and create swap you can either:

- Pass this in as a cloud-init config every time you create a VM
- Use a cloud-init directive baked into the image that will do this every time the VM is created:

```

echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
ephemeral0:
  table_type: mbr
  layout: [66, [33, 82]]
  overwrite: True
fs_setup:
- device: ephemeral0.1
  filesystem: ext4
- device: ephemeral0.2
  filesystem: swap
mounts:
- ["ephemeral0.1", "/mnt"]
- ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service,x-
systemd.device-timeout=2", "0", "0"]
EOF

```

14. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

**Note:** if you are migrating a specific virtual machine and do not wish to create a generalized image, skip the deprovision step

```

# sudo rm -f /var/log/waagent.log
# sudo cloud-init clean
# waagent -force -deprovision+user
# rm -f ~/.bash_history
# export HISTSIZE=0
# logout

```

15. Click Action -> **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## Next steps

You're now ready to use your CentOS Linux virtual hard disk to create new virtual machines in Azure. If this is the first time that you're uploading the .vhd file to Azure, see [Create a Linux VM from a custom disk](#).

# Prepare a Debian VHD for Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

## Prerequisites

This section assumes that you have already installed a Debian Linux operating system from an .iso file downloaded from the [Debian website](#) to a virtual hard disk. Multiple tools exist to create .vhd files; Hyper-V is only one example. For instructions using Hyper-V, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

## Installation notes

- See also [General Linux Installation Notes](#) for more tips on preparing Linux for Azure.
- The newer VHDX format is not supported in Azure. You can convert the disk to VHD format using Hyper-V Manager or the `convert-vhd` cmdlet.
- When installing the Linux system, it is recommended that you use standard partitions rather than LVM (often the default for many installations). This will avoid LVM name conflicts with cloned VMs, particularly if an OS disk ever needs to be attached to another VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks if preferred.
- Do not configure a swap partition on the OS disk. The Azure Linux agent can be configured to create a swap file on the temporary resource disk. More information can be found in the steps below.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD, you must ensure that the raw disk size is a multiple of 1MB before conversion. For more information, see [Linux Installation Notes](#).

## Use Azure-Manage to create Debian VHDs

There are tools available for generating Debian VHDs for Azure, such as the `azure-manage` scripts from [Credativ](#). This is the recommended approach versus creating an image from scratch. For example, to create a Debian 8 VHD run the following commands to download the `azure-manage` utility (and dependencies) and run the `azure_build_image` script:

```
# sudo apt-get update
# sudo apt-get install git qemu-utils mbr kpartx debootstrap

# sudo apt-get install python3-pip python3-dateutil python3-cryptography
# sudo pip3 install azure-storage azure-servicemanagement-legacy azure-common pytest pyyaml
# git clone https://github.com/credativ/azure-manage.git
# cd azure-manage
# sudo pip3 install .

# sudo azure_build_image --option release=jessie --option image_size_gb=30 --option image_prefix=debian-jessie-azure section
```

## Prepare a Debian image for Azure

You can create the base Azure Debian Cloud image with the [FAI cloud image builder](#).

(The following git clone and apt install commands were pulled from the Debian Cloud Images repo) Start by cloning the repo and installing dependencies:

```
$ git clone https://salsa.debian.org/cloud-team/debian-cloud-images.git
$ sudo apt install --no-install-recommends ca-certificates debsums dosfstools \
    fai-server fai-setup-storage make python3 python3-libcloud python3-marshmallow \
    python3-pytest python3-yaml qemu-utils udev
$ cd ./debian-cloud-images
```

(Optional) Customize the build by adding scripts (e.g. shell scripts) to `./config_space/scripts/AZURE`.

An example of a script to customize the image is:

```
$ mkdir -p ./config_space/scripts/AZURE
$ cat > ./config_space/scripts/AZURE/10-custom <<EOF
#!/bin/bash

\$ROOTCMD bash -c "echo test > /usr/local/share/testing"
EOF
$ sudo chmod 755 ./config_space/scripts/AZURE/10-custom
```

Note that it is important to prefix any commands you want to have customizing the image with `\$ROOTCMD` as this is aliased as `chroot $target`.

Build the Azure Debian 10 image:

```
$ make image_buster_azure_amd64
```

This will output a handful of files in the current directory, most notably the `image_buster_azure_amd64.raw` image file.

To convert the raw image to VHD for Azure, you can do the following:

```
rawdisk="image_buster_azure_amd64.raw"
vhddisk="image_buster_azure_amd64.vhd"

MB=$((1024*1024))
size=$(qemu-img info -f raw --output json "$rawdisk" | \
gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')

rounded_size=$(((($size+$MB-1)/$MB)*$MB))
rounded_size_adjusted=$((rounded_size + 512))

echo "Rounded Size Adjusted = $rounded_size_adjusted"

sudo qemu-img resize "$rawdisk" $rounded_size
qemu-img convert -f raw -o subformat=fixed,force_size -O vpc "$rawdisk" "$vhddisk"
```

This creates a VHD `image_buster_azure_amd64.vhd` with a rounded size to be able to copy it successfully to an Azure Disk.

Now we need to create the Azure resources for this image (this uses the `$rounded_size_adjusted` variable, so it should be from within the same shell process from above).

```

az group create -l $LOCATION -n $RG

az disk create \
    -n $DISK \
    -g $RG \
    -l $LOCATION \
    --for-upload --upload-size-bytes "$rounded_size_adjusted" \
    --sku standard_lrs --hyper-v-generation V1

ACCESS=$(az disk grant-access \
    -n $DISK -g $RG \
    --access-level write \
    --duration-in-seconds 86400 \
    --query accessSas -o tsv)

azcopy copy "$vhddisk" "$ACCESS" --blob-type PageBlob

az disk revoke-access -n $DISK -g $RG
az image create \
    -g $RG \
    -n $IMAGE \
    --os-type linux \
    --source $(az disk show \
        -g $RG \
        -n $DISK \
        --query id -o tsv)
az vm create \
    -g $RG \
    -n $VM \
    --ssh-key-value $SSH_KEY_VALUE \
    --public-ip-address-dns-name $VM \
    --image $(az image show \
        -g $RG \
        -n $IMAGE \
        --query id -o tsv)

```

#### NOTE

If the bandwidth from your local machine to the Azure Disk is causing a long time to process the upload with azcopy, you can use an Azure VM jumpbox to speed up the process. Here's how this can be done:

1. Create a tarball of the VHD on your local machine:

```
tar -czvf ./image_buster_azure_amd64.vhd.tar.gz ./image_buster_azure_amd64.vhd .
```

2. Create an Azure Linux VM (distro of your choice). Make sure that you create it with a large enough disk to hold the extracted VHD!
3. Download the azcopy utility to the Azure Linux VM. It can be retrieved from [here](#).
4. Copy the tarball to the VM: 

```
scp ./image_buster_azure_amd64.vhd.tar.gz <vm>:~ .
```
5. On the VM, extract the VHD: 

```
tar -xf ./image_buster_azure_amd64.vhd.tar.gz
```

 (this will take a bit of time given the size of the file).
6. Finally on the VM, copy the VHD to the Azure Disk with `azcopy` (the command from above).

**Next steps:** You're now ready to use your Debian Linux virtual hard disk to create new virtual machines in Azure. If this is the first time that you're uploading the .vhd file to Azure, see [Create a Linux VM from a custom disk](#).

# Using a prebuilt Flatcar image for Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You can download prebuilt Azure virtual hard disk images of Flatcar Container Linux for each of the Flatcar supported channels:

- [stable](#)
- [beta](#)
- [alpha](#)
- [LTS](#)

This image is already fully set up and optimized to run on Azure. You only need to decompress it.

## Building your own Flatcar-based virtual machine for Azure

Alternatively, you can choose to build your own Flatcar Container Linux image.

On any linux based machine, follow the instructions detailed in the [Flatcar Container Linux developer SDK guide](#).

When running the `image_to_vm.sh` script, make sure you pass `--format=azure` to create an Azure virtual hard disk.

## Next steps

Once you have the .vhd file, you can use the resulting file to create new virtual machines in Azure. If this is the first time that you're uploading a .vhd file to Azure, see [Create a Linux VM from a custom disk](#).

# Introduction to FreeBSD on Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article provides an overview of running a FreeBSD virtual machine in Azure.

## Overview

FreeBSD for Microsoft Azure is an advanced computer operating system used to power modern servers, desktops, and embedded platforms.

Microsoft Corporation is making images of FreeBSD available on Azure with the [Azure VM Guest Agent](#) pre-configured. Currently, the following FreeBSD versions are offered as images by Microsoft:

- FreeBSD 10.4 on the Azure Marketplace
- FreeBSD 11.2 on the Azure Marketplace
- FreeBSD 11.3 on the Azure Marketplace
- FreeBSD 12.0 on the Azure Marketplace

The following FreeBSD versions also include the [Azure VM Guest Agent](#), however, they are offered as images by the FreeBSD Foundation:

- FreeBSD 11.4 on the Azure Marketplace
- FreeBSD 12.2 on the Azure Marketplace
- FreeBSD 13.0 on the Azure Marketplace

The agent is responsible for communication between the FreeBSD VM and the Azure fabric for operations such as provisioning the VM on first use (user name, password or SSH key, host name, etc.) and enabling functionality for selective VM extensions.

As for future versions of FreeBSD, the strategy is to stay current and make the latest releases available shortly after they are published by the FreeBSD release engineering team.

### Create a FreeBSD VM through Azure CLI on FreeBSD

First you need to install [Azure CLI](#) though following command on a FreeBSD machine.

```
curl -L https://aka.ms/InstallAzureCli | bash
```

If bash is not installed on your FreeBSD machine, run following command before the installation.

```
sudo pkg install bash
```

If Python is not installed on your FreeBSD machine, run following commands before the installation.

```
sudo pkg install python38
cd /usr/local/bin
sudo rm /usr/local/bin/python
sudo ln -s /usr/local/bin/python3.8 /usr/local/bin/python
```

During the installation, you are asked

Modify profile to update your \$PATH and enable shell/tab completion now? (Y/n) . If you answer `y` and enter `/etc/rc.conf` as a path to an rc file to update , you may meet the problem

ERROR: [Errno 13] Permission denied . To resolve this problem, you should grant the write right to current user against the file `/etc/rc.conf` .

Now you can sign in to Azure and create your FreeBSD VM. Below is an example to create a FreeBSD 11.0 VM. You can also add the parameter `--public-ip-address-dns-name` with a globally unique DNS name for a newly created Public IP.

```
az login
az group create --name myResourceGroup --location eastus
az vm create --name myFreeBSD11 \
    --resource-group myResourceGroup \
    --image MicrosoftOSTC:FreeBSD:11.0:latest \
    --admin-username azureuser \
    --generate-ssh-keys
```

Then you can sign in to your FreeBSD VM through the ip address that printed in the output of above deployment.

```
ssh azureuser@xx.xx.xx.xx -i /etc/ssh/ssh_host_rsa_key
```

## VM extensions for FreeBSD

Following are supported VM extensions in FreeBSD.

### VMAccess

The [VMAccess](#) extension can:

- Reset the password of the original sudo user.
- Create a new sudo user with the password specified.
- Set the public host key with the key given.
- Reset the public host key provided during VM provisioning if the host key is not provided.
- Open the SSH port (22) and restore the `sshd_config` if `reset_ssh` is set to true.
- Remove the existing user.
- Check disks.
- Repair an added disk.

### CustomScript

The [CustomScript](#) extension can:

- If provided, download the customized scripts from Azure Storage or external public storage (for example, GitHub).
- Run the entry point script.
- Support inline commands.
- Convert Windows-style newline in shell and Python scripts automatically.
- Remove BOM in shell and Python scripts automatically.
- Protect sensitive data in `CommandToExecute`.

#### NOTE

FreeBSD VM only supports CustomScript version 1.x by now.

## Authentication: user names, passwords, and SSH keys

When you're creating a FreeBSD virtual machine by using the Azure portal, you must provide a user name, password, or SSH public key. User names for deploying a FreeBSD virtual machine on Azure must not match names of system accounts (UID < 100) already present in the virtual machine ("root", for example). Currently, only the RSA SSH key is supported. A multiline SSH key must begin with `----- BEGIN SSH2 PUBLIC KEY -----` and end with `----- END SSH2 PUBLIC KEY -----`.

## Obtaining superuser privileges

The user account that is specified during virtual machine instance deployment on Azure is a privileged account. The package of sudo was installed in the published FreeBSD image. After you're logged in through this user account, you can run commands as root by using the command syntax.

```
$ sudo <COMMAND>
```

You can optionally obtain a root shell by using `sudo -s`.

## Known issues

The [Azure VM Guest Agent](#) version 2.2.2 has a [known issue](#) that causes the provision failure for FreeBSD VM on Azure. The fix was captured by [Azure VM Guest Agent](#) version 2.2.3 and later releases.

## Next steps

- Go to [Azure Marketplace](#) to create a FreeBSD VM.

# Prepare an Oracle Linux virtual machine for Azure

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article assumes that you have already installed an Oracle Linux operating system to a virtual hard disk. Multiple tools exist to create .vhdx files, for example a virtualization solution such as Hyper-V. For instructions, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

## Oracle Linux installation notes

- Please see also [General Linux Installation Notes](#) for more tips on preparing Linux for Azure.
- Hyper-V and Azure support Oracle Linux with either the Unbreakable Enterprise Kernel (UEK) or the Red Hat Compatible Kernel.
- Oracle's UEK2 is not supported on Hyper-V and Azure as it does not include the required drivers.
- The VHDX format is not supported in Azure, only **fixed VHD**. You can convert the disk to VHD format using Hyper-V Manager or the convert-vhd cmdlet.
- **Kernel support for mounting UDF file systems is required.** At first boot on Azure the provisioning configuration is passed to the Linux VM via UDF-formatted media that is attached to the guest. The Azure Linux agent must be able to mount the UDF file system to read its configuration and provision the VM.
- When installing the Linux system it is recommended that you use standard partitions rather than LVM (often the default for many installations). This will avoid LVM name conflicts with cloned VMs, particularly if an OS disk ever needs to be attached to another VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks if preferred.
- Linux kernel versions earlier than 2.6.37 don't support NUMA on Hyper-V with larger VM sizes. This issue primarily impacts older distributions using the upstream Red Hat 2.6.32 kernel, and was fixed in Oracle Linux 6.6 and later
- Do not configure a swap partition on the OS disk. More information about this can be found in the steps below.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1MB before conversion. See [Linux Installation Notes](#) for more information.
- Make sure that the `Addons` repository is enabled. Edit the file `/etc/yum.repos.d/public-yum-ol6.repo` (Oracle Linux 6) or `/etc/yum.repos.d/public-yum-ol7.repo` (Oracle Linux 7), and change the line `enabled=0` to `enabled=1` under `[ol6_addons]` or `[ol7_addons]` in this file.

## Oracle Linux 6.4 and later

You must complete specific configuration steps in the operating system for the virtual machine to run in Azure.

1. In the center pane of Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open the window for the virtual machine.
3. Uninstall NetworkManager by running the following command:

```
# sudo rpm -e --nodeps NetworkManager
```

**Note:** If the package is not already installed, this command will fail with an error message. This is

expected.

4. Create a file named **network** in the `/etc/sysconfig/` directory that contains the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

5. Create a file named **ifcfg-eth0** in the `/etc/sysconfig/network-scripts/` directory that contains the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no
```

6. Modify udev rules to avoid generating static rules for the Ethernet interface(s). These rules can cause problems when cloning a virtual machine in Microsoft Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules  
# sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

7. Ensure the network service will start at boot time by running the following command:

```
# chkconfig network on
```

8. Install `python-pyasn1` by running the following command:

```
# sudo yum install python-pyasn1
```

9. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this open `/boot/grub/menu.lst` in a text editor and ensure that the kernel includes the following parameters:

```
console=ttyS0 earlyprintk=ttyS0
```

This will ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues.

In addition to the above, it is recommended to *remove* the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port.

The `crashkernel` option may be left configured if desired, but note that this parameter will reduce the amount of available memory in the VM by 128MB or more, which may be problematic on the smaller VM sizes.

10. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.

11. Install the Azure Linux Agent by running the following command. The latest version is 2.0.15.

```
# sudo yum install WALinuxAgent
```

Note that installing the WALinuxAgent package will remove the NetworkManager and NetworkManager-gnome packages if they were not already removed as described in step 2.

12. Do not create swap space on the OS disk.

The Azure Linux Agent can automatically configure swap space using the local resource disk that is attached to the VM after provisioning on Azure. Note that the local resource disk is a *temporary* disk, and might be emptied when the VM is deprovisioned. After installing the Azure Linux Agent (see previous step), modify the following parameters in /etc/waagent.conf appropriately:

```
ResourceDisk.Format=y
ResourceDisk.Filesystem=ext4
ResourceDisk.MountPoint=/mnt/resource
ResourceDisk.EnableSwap=y
ResourceDisk.SwapSizeMB=2048    ## NOTE: set this to whatever you need it to be.
```

13. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# sudo waagent -force -deprovision
# export HISTSIZE=0
# logout
```

14. Click Action -> Shut Down in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## Oracle Linux 7.0 and later

### Changes in Oracle Linux 7

Preparing an Oracle Linux 7 virtual machine for Azure is very similar to Oracle Linux 6, however there are several important differences worth noting:

- Azure supports Oracle Linux with either the Unbreakable Enterprise Kernel (UEK) or the Red Hat Compatible Kernel. Oracle Linux with UEK is recommended.
- The NetworkManager package no longer conflicts with the Azure Linux agent. This package is installed by default and we recommend that it is not removed.
- GRUB2 is now used as the default bootloader, so the procedure for editing kernel parameters has changed (see below).
- XFS is now the default file system. The ext4 file system can still be used if desired.

### Configuration steps

1. In Hyper-V Manager, select the virtual machine.
2. Click Connect to open a console window for the virtual machine.
3. Create a file named **network** in the `/etc/sysconfig/` directory that contains the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

4. Create a file named `ifcfg-eth0` in the `/etc/sysconfig/network-scripts/` directory that contains the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no
```

5. Modify udev rules to avoid generating static rules for the Ethernet interface(s). These rules can cause problems when cloning a virtual machine in Microsoft Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules
```

6. Ensure the network service will start at boot time by running the following command:

```
# sudo chkconfig network on
```

7. Install the `python-pyasn1` package by running the following command:

```
# sudo yum install python-pyasn1
```

8. Run the following command to clear the current yum metadata and install any updates:

```
# sudo yum clean all  
# sudo yum -y update
```

9. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this open `/etc/default/grub` in a text editor and edit the `GRUB_CMDLINE_LINUX` parameter, for example:

```
GRUB_CMDLINE_LINUX="console=ttyS0 earlyprintk=ttyS0 net.ifnames=0"
```

This will also ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues. It also turns off the naming conventions for NICs in Oracle Linux 7 with the Unbreakable Enterprise Kernel. In addition to the above, it is recommended to *remove* the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port.

The `crashkernel` option may be left configured if desired, but note that this parameter will reduce the amount of available memory in the VM by 128MB or more, which may be problematic on the smaller VM sizes.

10. Once you are done editing "/etc/default/grub" per above, run the following command to rebuild the grub configuration:

```
# sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

11. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.

12. Install the Azure Linux Agent and dependencies:

```
sudo yum install WALinuxAgent  
sudo systemctl enable waagent
```

13. Install cloud-init to handle the provisioning

```
yum install -y cloud-init cloud-utils-growpart gdisk hyperv-daemons

# Configure waagent for cloud-init
sed -i 's/Provisioning.UseCloudInit=n/Provisioning.UseCloudInit=y/g' /etc/waagent.conf
sed -i 's/Provisioning.Enabled=n/Provisioning.Enabled=y/g' /etc/waagent.conf

echo "Adding mounts and disk_setup to init stage"
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg

echo "Allow only Azure datasource, disable fetching network setting via IMDS"
cat > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg <<EOF
datasource_list: [ Azure ]
datasource:
    Azure:
        apply_network_config: False
EOF

if [[ -f /mnt/resource/swapfile ]]; then
echo Removing swapfile - RHEL uses a swapfile by default
swapoff /mnt/resource/swapfile
rm /mnt/resource/swapfile -f
fi

echo "Add console log file"
cat >> /etc/cloud/cloud.cfg.d/05_logging.cfg <<EOF

# This tells cloud-init to redirect its stdout and stderr to
# 'tee -a /var/log/cloud-init-output.log' so the user can see output
# there without needing to look on the console.
output: {all: '| tee -a /var/log/cloud-init-output.log'}
EOF
```

14. Swap configuration Do not create swap space on the operating system disk.

Previously, the Azure Linux Agent was used automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However this is now handled by cloud-init, you **must not** use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf
```

If you want mount, format and create swap you can either:

- Pass this in as a cloud-init config every time you create a VM
- Use a cloud-init directive baked into the image that will do this every time the VM is created:

```
echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
  ephemeral0:
    table_type: mbr
    layout: [66, [33, 82]]
    overwrite: True
fs_setup:
  - device: ephemeral0.1
    filesystem: ext4
  - device: ephemeral0.2
    filesystem: swap
mounts:
  - ["ephemeral0.1", "/mnt"]
  - ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service,x-
systemd.device-timeout=2", "0", "0"]
EOF
```

15. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

**Note:** if you are migrating a specific virtual machine and do not wish to create a generalized image, skip the deprovision step

```
# sudo cloud-init clean
# sudo rm -f /var/log/waagent.log

# waagent -force -deprovision+user
# rm -f ~/.bash_history

# export HISTSIZE=0

# logout
```

16. Click Action -> **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## Next steps

You're now ready to use your Oracle Linux .vhd to create new virtual machines in Azure. If this is the first time that you're uploading the .vhd file to Azure, see [Create a Linux VM from a custom disk](#).

# Create and Upload an OpenBSD disk image to Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to create and upload a virtual hard disk (VHD) that contains the OpenBSD operating system. After you upload it, you can use it as your own image to create a virtual machine (VM) in Azure through Azure CLI.

## Prerequisites

This article assumes that you have the following items:

- **An Azure subscription** - If you don't have an account, you can create one in just a couple of minutes. If you have an MSDN subscription, see [Monthly Azure credit for Visual Studio subscribers](#). Otherwise, learn how to [create a free trial account](#).
- **Azure CLI** - Make sure you have the latest [Azure CLI](#) installed and logged in to your Azure account with [az login](#).
- **OpenBSD operating system installed in a .vhd file** - A supported OpenBSD operating system ([6.6 version AMD64](#)) must be installed to a virtual hard disk. Multiple tools exist to create .vhd files. For example, you can use a virtualization solution such as Hyper-V to create the .vhd file and install the operating system. For instructions about how to install and use Hyper-V, see [Install Hyper-V and create a virtual machine](#).

## Prepare OpenBSD image for Azure

On the VM where you installed the OpenBSD operating system 6.1, which added Hyper-V support, complete the following procedures:

1. If DHCP is not enabled during installation, enable the service as follows:

```
echo dhcp > /etc/hostname.hvn0
```

2. Set up a serial console as follows:

```
echo "stty com0 115200" >> /etc/boot.conf
echo "set tty com0" >> /etc/boot.conf
```

3. Configure Package installation as follows:

```
echo "https://ftp.openbsd.org/pub/OpenBSD" > /etc/installurl
```

4. By default, the `root` user is disabled on virtual machines in Azure. Users can run commands with elevated privileges by using the `doas` command on OpenBSD VM. Doas is enabled by default. For more information, see [doas.conf](#).
5. Install and configure prerequisites for the Azure Agent as follows:

```
pkg_add py-setuptools openssl git
ln -sf /usr/local/bin/python2.7 /usr/local/bin/python
ln -sf /usr/local/bin/python2.7-2to3 /usr/local/bin/2to3
ln -sf /usr/local/bin/python2.7-config /usr/local/bin/python-config
ln -sf /usr/local/bin/pydoc2.7 /usr/local/bin/pydoc
```

6. The latest release of the Azure agent can always be found on [GitHub](#). Install the agent as follows:

```
git clone https://github.com/Azure/WALinuxAgent
cd WALinuxAgent
python setup.py install
waagent -register-service
```

#### IMPORTANT

After you install Azure Agent, it's a good idea to verify that it's running as follows:

```
ps auxw | grep waagent
root      79309  0.0  1.5  9184 15356 p1  S      4:11PM   0:00.46 python /usr/local/sbin/waagent
-daemon (python2.7)
cat /var/log/waagent.log
```

7. Deprovision the system to clean it and make it suitable for reprovisioning. The following command also deletes the last provisioned user account and the associated data:

```
waagent -deprovision+user -force
```

Now you can shut down your VM.

## Prepare the VHD

The VHDX format is not supported in Azure, only **fixed VHD**. You can convert the disk to fixed VHD format using Hyper-V Manager or the PowerShell `convert-vhd` cmdlet. An example is as following.

```
Convert-VHD OpenBSD61.vhdx OpenBSD61.vhd -VHDTType Fixed
```

## Create storage resources and upload

First, create a resource group with `az group create`. The following example creates a resource group named `myResourceGroup` in the `eastus` location:

```
az group create --name myResourceGroup --location eastus
```

To upload your VHD, create a storage account with `az storage account create`. Storage account names must be unique, so provide your own name. The following example creates a storage account named `mystorageaccount`.

```
az storage account create --resource-group myResourceGroup \
--name mystorageaccount \
--location eastus \
--sku Premium_LRS
```

To control access to the storage account, obtain the storage key with [az storage account keys list](#) as follows:

```
STORAGE_KEY=$(az storage account keys list \
--resource-group myResourceGroup \
--account-name mystorageaccount \
--query "[?keyName=='key1'] | [0].value" -o tsv)
```

To logically separate the VHDs you upload, create a container within the storage account with [az storage container create](#):

```
az storage container create \
--name vhds \
--account-name mystorageaccount \
--account-key ${STORAGE_KEY}
```

Finally, upload your VHD with [az storage blob upload](#) as follows:

```
az storage blob upload \
--container-name vhds \
--file ./OpenBSD61.vhd \
--name OpenBSD61.vhd \
--account-name mystorageaccount \
--account-key ${STORAGE_KEY}
```

## Create VM from your VHD

You can create a VM with a [sample script](#) or directly with [az vm create](#). To specify the OpenBSD VHD you uploaded, use the `--image` parameter as follows:

```
az vm create \
--resource-group myResourceGroup \
--name myOpenBSD61 \
--image "https://mystorageaccount.blob.core.windows.net/vhds/OpenBSD61.vhd" \
--os-type linux \
--admin-username azureuser \
--ssh-key-value ~/.ssh/id_rsa.pub
```

Obtain the IP address for your OpenBSD VM with [az vm list-ip-addresses](#) as follows:

```
az vm list-ip-addresses --resource-group myResourceGroup --name myOpenBSD61
```

Now you can SSH to your OpenBSD VM as normal:

```
ssh azureuser@<ip address>
```

## Next steps

If you want to know more about Hyper-V support on OpenBSD6.1, read [OpenBSD 6.1](#) and [hyperv.4](#).

If you want to create a VM from managed disk, read [az disk](#).

# Prepare a Red Hat-based virtual machine for Azure

9/21/2022 • 32 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

In this article, you will learn how to prepare a Red Hat Enterprise Linux (RHEL) virtual machine for use in Azure. The versions of RHEL that are covered in this article are 6.7+ and 7.1+. The hypervisors for preparation that are covered in this article are Hyper-V, kernel-based virtual machine (KVM), and VMware. For more information about eligibility requirements for participating in Red Hat's Cloud Access program, see [Red Hat's Cloud Access website](#) and [Running RHEL on Azure](#). For ways to automate building RHEL images, see [Azure Image Builder](#).

## Hyper-V Manager

This section shows you how to prepare a [RHEL 6](#), [RHEL 7](#), or [RHEL 8](#) virtual machine using Hyper-V Manager.

### Prerequisites

This section assumes that you have already obtained an ISO file from the Red Hat website and installed the RHEL image to a virtual hard disk (VHD). For more details about how to use Hyper-V Manager to install an operating system image, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

### RHEL installation notes

- Azure does not support the VHDX format. Azure supports only fixed VHD. You can use Hyper-V Manager to convert the disk to VHD format, or you can use the convert-vhd cmdlet. If you use VirtualBox, select **Fixed size** as opposed to the default dynamically allocated option when you create the disk.
- Azure supports Gen1 (BIOS boot) & Gen2 (UEFI boot) Virtual machines.
- The maximum size that's allowed for the VHD is 1,023 GB.
- Logical Volume Manager (LVM) is supported and may be used on the OS disk or data disks in Azure virtual machines. However, in general it is recommended to use standard partitions on the OS disk rather than LVM. This practice will avoid LVM name conflicts with cloned virtual machines, particularly if you ever need to attach an operating system disk to another identical virtual machine for troubleshooting. See also [LVM](#) and [RAID](#) documentation.
- **Kernel support for mounting Universal Disk Format (UDF) file systems is required.** At first boot on Azure, the UDF-formatted media that is attached to the guest passes the provisioning configuration to the Linux virtual machine. The Azure Linux Agent must be able to mount the UDF file system to read its configuration and provision the virtual machine, without this, provisioning will fail!
- Do not configure a swap partition on the operating system disk. More information about this can be found in the following steps.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1MB before conversion. More details can be found in the steps below. See also [Linux Installation Notes](#) for more information.

### RHEL 6 using Hyper-V Manager

1. In Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open a console window for the virtual machine.
3. In RHEL 6, NetworkManager can interfere with the Azure Linux agent. Uninstall this package by running

the following command:

```
# sudo rpm -e --nodeps NetworkManager
```

4. Create or edit the `/etc/sysconfig/network` file, and add the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

5. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no
```

6. Move (or remove) the udev rules to avoid generating static rules for the Ethernet interface. These rules cause problems when you clone a virtual machine in Microsoft Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules  
  
# sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

7. Ensure that the network service will start at boot time by running the following command:

```
# sudo chkconfig network on
```

8. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# sudo subscription-manager register --auto-attach --username=XXX --password=XXX
```

9. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository. Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-6-server-extras-rpms
```

10. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this modification, open `/boot/grub/menu.lst` in a text editor, and ensure that the default kernel includes the following parameters:

```
console=ttyS0 earlyprintk=ttyS0
```

This will also ensure that all console messages are sent to the first serial port, which can assist Azure support with debugging issues.

In addition, we recommended that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more. This configuration might be problematic on smaller virtual machine sizes.

11. Ensure that the secure shell (SSH) server is installed and configured to start at boot time, which is usually the default. Modify `/etc/ssh/sshd_config` to include the following line:

```
ClientAliveInterval 180
```

12. Install the Azure Linux Agent by running the following command:

```
# sudo yum install WALinuxAgent  
  
# sudo chkconfig waagent on
```

Installing the WALinuxAgent package removes the NetworkManager and NetworkManager-gnome packages if they were not already removed in step 3.

13. Do not create swap space on the operating system disk.

The Azure Linux Agent can automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. Note that the local resource disk is a temporary disk and that it might be emptied if the virtual machine is deprovisioned. After you install the Azure Linux Agent in the previous step, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=y  
ResourceDisk.Filesystem=ext4  
ResourceDisk.MountPoint=/mnt/resource  
ResourceDisk.EnableSwap=y  
ResourceDisk.SwapSizeMB=2048 ## NOTE: set this to whatever you need it to be.
```

14. Unregister the subscription (if necessary) by running the following command:

```
# sudo subscription-manager unregister
```

15. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# Note: if you are migrating a specific virtual machine and do not wish to create a generalized  
image,  
# skip the deprovision step  
# sudo waagent -force -deprovision  
  
# export HISTSIZE=0  
  
# logout
```

16. Click Action > **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## RHEL 7 using Hyper-V Manager

1. In Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open a console window for the virtual machine.
3. Create or edit the `/etc/sysconfig/network` file, and add the following text:

```
NETWORKING=yes  
HOSTNAME=localhost.localdomain
```

4. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0  
ONBOOT=yes  
BOOTPROTO=dhcp  
TYPE=Ethernet  
USERCTL=no  
PEERDNS=yes  
IPV6INIT=no  
PERSISTENT_DHCLIENT=yes  
NM_CONTROLLED=yes
```

5. Ensure that the network service will start at boot time by running the following command:

```
# sudo systemctl enable network
```

6. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# sudo subscription-manager register --auto-attach --username=XXX --password=XXX
```

7. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure.

To do this modification, open `/etc/default/grub` in a text editor, and edit the `GRUB_CMDLINE_LINUX` parameter. For example:

```
GRUB_CMDLINE_LINUX="console=tty1 console=ttyS0,115200n8 earlyprintk=ttyS0,115200 earlyprintk=ttyS0  
net.ifnames=0"  
GRUB_TERMINAL_OUTPUT="serial console"  
GRUB_SERIAL_COMMAND="serial --speed=115200 --unit=0 --word=8 --parity=no --stop=1"
```

This will also ensure that all console messages are sent to the first serial port and enable interaction with the serial console, which can assist Azure support with debugging issues. This configuration also turns off the new RHEL 7 naming conventions for NICs.

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

8. After you are done editing `/etc/default/grub`, run the following command to rebuild the grub configuration:

```
# sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

#### NOTE

If uploading an UEFI enabled VM, the command to update grub is

```
grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg .
```

9. Ensure that the SSH server is installed and configured to start at boot time, which is usually the default.

Modify `/etc/ssh/sshd_config` to include the following line:

```
ClientAliveInterval 180
```

10. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository.

Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-7-server-extras-rpms
```

11. Install the Azure Linux Agent, cloud-init and other necessary utilities by running the following command:

```
# sudo yum install -y WALinuxAgent cloud-init cloud-utils-growpart gdisk hyperv-daemons  
  
# sudo systemctl enable waagent.service  
# sudo systemctl enable cloud-init.service
```

12. Configure cloud-init to handle the provisioning:

- Configure waagent for cloud-init:

```
sed -i 's/Provisioning.Agent=auto/Provisioning.Agent=cloud-init/g' /etc/waagent.conf  
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf  
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf
```

#### NOTE

If you are migrating a specific virtual machine and do not wish to create a generalized image, set

```
Provisioning.Agent=disabled
```

 on the `/etc/waagent.conf` config.

- Configure mounts:

```
echo "Adding mounts and disk_setup to init stage"  
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg  
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg  
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg  
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg
```

- Configure Azure datasource:

```
echo "Allow only Azure datasource, disable fetching network setting via IMDS"
cat > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg <<EOF
datasource_list: [ Azure ]
datasource:
  Azure:
    apply_network_config: False
EOF
```

- a. If configured, remove existing swapfile:

```
if [[ -f /mnt/resource/swapfile ]]; then
  echo "Removing swapfile" #RHEL uses a swapfile by default
  swapoff /mnt/resource/swapfile
  rm /mnt/resource/swapfile -f
fi
```

- a. Configure cloud-init logging:

```
echo "Add console log file"
cat >> /etc/cloud/cloud.cfg.d/05_logging.cfg <<EOF

# This tells cloud-init to redirect its stdout and stderr to
# 'tee -a /var/log/cloud-init-output.log' so the user can see output
# there without needing to look on the console.
output: {all: '| tee -a /var/log/cloud-init-output.log'}
EOF
```

### 13. Swap configuration Do not create swap space on the operating system disk.

Previously, the Azure Linux Agent was used automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However this is now handled by cloud-init, you **must not** use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=n
ResourceDisk.EnableSwap=n
```

If you want mount, format and create swap you can either:

- Pass this in as a cloud-init config every time you create a VM through customdata. This is the recommended method.
- Use a cloud-init directive baked into the image that will do this every time the VM is created.

```

echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
  ephemeral0:
    table_type: mbr
    layout: [66, [33, 82]]
    overwrite: True
fs_setup:
  - device: ephemeral0.1
    filesystem: ext4
  - device: ephemeral0.2
    filesystem: swap
mounts:
  - ["ephemeral0.1", "/mnt"]
  - ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service,x-
systemd.device-timeout=2", "0", "0"]
EOF

```

14. If you want to unregister the subscription, run the following command:

```
# sudo subscription-manager unregister
```

15. Deprovision

Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

**Caution**

If you are migrating a specific virtual machine and do not wish to create a generalized image, skip the deprovision step. Running the command `waagent -force -deprovision+user` will render the source machine unusable, this step is intended only to create a generalized image.

```

# sudo rm -f /var/log/waagent.log
# sudo cloud-init clean
# waagent -force -deprovision+user
# rm -f ~/.bash_history
# export HISTSIZE=0
# logout

```

16. Click **Action > Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## RHEL 8 using Hyper-V Manager

- In Hyper-V Manager, select the virtual machine.
- Click **Connect** to open a console window for the virtual machine.
- Ensure that the Network Manager service will start at boot time by running the following command:

```
# sudo systemctl enable NetworkManager.service
```

- Configure the network interface to automatically start at boot and use DHCP:

```
# nmcli con mod eth0 connection.autoconnect yes ipv4.method auto
```

5. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# sudo subscription-manager register --auto-attach --username=XXX --password=XXX
```

6. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure and enable the serial console.

- a. Remove current GRUB parameters:

```
# grub2-editenv - unset kernelopts
```

- a. Edit `/etc/default/grub` in a text editor, and add the following parameters:

```
GRUB_CMDLINE_LINUX="console=tty1 console=ttyS0,115200n8 earlyprintk=ttyS0,115200 earlyprintk=ttyS0  
net.ifnames=0"  
GRUB_TERMINAL_OUTPUT="serial console"  
GRUB_SERIAL_COMMAND="serial --speed=115200 --unit=0 --word=8 --parity=no --stop=1"
```

This will also ensure that all console messages are sent to the first serial port and enable interaction with the serial console, which can assist Azure support with debugging issues. This configuration also turns off the new naming conventions for NICs.

- a. Additionally, we recommend that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

7. After you are done editing `/etc/default/grub`, run the following command to rebuild the grub configuration:

```
# sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

And for an UEFI enabled VM, run the following command:

```
# sudo grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

8. Ensure that the SSH server is installed and configured to start at boot time, which is usually the default. Modify `/etc/ssh/sshd_config` to include the following line:

```
ClientAliveInterval 180
```

9. Install the Azure Linux Agent, cloud-init and other necessary utilities by running the following command:

```
# sudo yum install -y WALinuxAgent cloud-init cloud-utils-growpart gdisk hyperv-daemons  
  
# sudo systemctl enable waagent.service  
# sudo systemctl enable cloud-init.service
```

## 10. Configure cloud-init to handle the provisioning:

### a. Configure waagent for cloud-init:

```
sed -i 's/Provisioning.Agent=auto/Provisioning.Agent=cloud-init/g' /etc/waagent.conf  
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf  
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf
```

#### NOTE

If you are migrating a specific virtual machine and do not wish to create a generalized image, set `Provisioning.Agent=disabled` on the `/etc/waagent.conf` config.

### a. Configure mounts:

```
echo "Adding mounts and disk_setup to init stage"  
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg  
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg  
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg  
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg
```

### a. Configure Azure datasource:

```
echo "Allow only Azure datasource, disable fetching network setting via IMDS"  
cat > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg <<EOF  
datasource_list: [ Azure ]  
datasource:  
    Azure:  
        apply_network_config: False  
EOF
```

### a. If configured, remove existing swapfile:

```
if [[ -f /mnt/resource/swapfile ]]; then  
    echo "Removing swapfile" #RHEL uses a swapfile by default  
    swapoff /mnt/resource/swapfile  
    rm /mnt/resource/swapfile -f  
fi
```

### a. Configure cloud-init logging:

```
echo "Add console log file"  
cat >> /etc/cloud/cloud.cfg.d/05_logging.cfg <<EOF  
  
# This tells cloud-init to redirect its stdout and stderr to  
# 'tee -a /var/log/cloud-init-output.log' so the user can see output  
# there without needing to look on the console.  
output: {all: '| tee -a /var/log/cloud-init-output.log'}  
EOF
```

## 11. Swap configuration

Do not create swap space on the operating system disk.

Previously, the Azure Linux Agent was used automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However this is now handled by cloud-init, you **must not** use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=n  
ResourceDisk.EnableSwap=n
```

- Pass this in as a cloud-init config every time you create a VM through customdata. This is the recommended method.
- Use a cloud-init directive baked into the image that will do this every time the VM is created.

```
echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>  
/etc/systemd/system.conf  
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF  
#cloud-config  
# Generated by Azure cloud image build  
disk_setup:  
  ephemeral0:  
    table_type: mbr  
    layout: [66, [33, 82]]  
    overwrite: True  
fs_setup:  
  - device: ephemeral0.1  
    filesystem: ext4  
  - device: ephemeral0.2  
    filesystem: swap  
mounts:  
  - ["ephemeral0.1", "/mnt"]  
  - ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.device-timeout=2,x-  
systemd.requires=cloud-init.service", "0", "0"]  
EOF
```

## 12. If you want to unregister the subscription, run the following command:

```
# sudo subscription-manager unregister
```

## 13. Deprovision

Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# sudo cloud-init clean  
# waagent -force -deprovision+user  
# rm -f ~/.bash_history  
# sudo rm -f /var/log/waagent.log  
# export HISTSIZE=0  
# logout
```

### Caution

If you are migrating a specific virtual machine and do not wish to create a generalized image, skip the deprovision step. Running the command `waagent -force -deprovision+user` will render the source machine unusable, this step is intended only to create a generalized image.

## 14. Click Action > Shut Down in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to](#)

## KVM

This section shows you how to use KVM to prepare a [RHEL 6](#) or [RHEL 7](#) distro to upload to Azure.

### RHEL 6 using KVM

1. Download the KVM image of RHEL 6 from the Red Hat website.
2. Set a root password.

Generate an encrypted password, and copy the output of the command:

```
# openssl passwd -1 changeme
```

Set a root password with guestfish:

```
# guestfish --rw -a <image-name>
> <fs> run
> <fs> list-filesystems
> <fs> mount /dev/sda1 /
> <fs> vi /etc/shadow
> <fs> exit
```

Change the second field of the root user from "!!" to the encrypted password.

3. Create a virtual machine in KVM from the qcow2 image. Set the disk type to **qcow2**, and set the virtual network interface device model to **virtio**. Then, start the virtual machine, and sign in as root.
4. Create or edit the `/etc/sysconfig/network` file, and add the following text:

```
NETWORKING=yes
HOSTNAME=localhost.localdomain
```

5. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
TYPE=Ethernet
USERCTL=no
PEERDNS=yes
IPV6INIT=no
```

6. Move (or remove) the udev rules to avoid generating static rules for the Ethernet interface. These rules cause problems when you clone a virtual machine in Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules
# sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

7. Ensure that the network service will start at boot time by running the following command:

```
# chkconfig network on
```

8. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# subscription-manager register --auto-attach --username=XXX --password=XXX
```

9. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this configuration, open `/boot/grub/menu.lst` in a text editor, and ensure that the default kernel includes the following parameters:

```
console=ttyS0 earlyprintk=ttyS0
```

This will also ensure that all console messages are sent to the first serial port, which can assist Azure support with debugging issues.

In addition, we recommend that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

10. Add Hyper-V modules to initramfs:

Edit `/etc/dracut.conf`, and add the following content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild initramfs:

```
# dracut -f -v
```

11. Uninstall cloud-init:

```
# yum remove cloud-init
```

12. Ensure that the SSH server is installed and configured to start at boot time:

```
# chkconfig sshd on
```

Modify `/etc/ssh/sshd_config` to include the following lines:

```
PasswordAuthentication yes  
ClientAliveInterval 180
```

13. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository.

Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-6-server-extras-rpms
```

14. Install the Azure Linux Agent by running the following command:

```
# yum install WALinuxAgent  
# chkconfig waagent on
```

15. The Azure Linux Agent can automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. Note that the local resource disk is a temporary disk, and it might be emptied if the virtual machine is deprovisioned. After you install the Azure Linux Agent in the previous step, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=y  
ResourceDisk.Filesystem=ext4  
ResourceDisk.MountPoint=/mnt/resource  
ResourceDisk.EnableSwap=y  
ResourceDisk.SwapSizeMB=2048    ## NOTE: set this to whatever you need it to be.
```

16. Unregister the subscription (if necessary) by running the following command:

```
# subscription-manager unregister
```

17. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# Note: if you are migrating a specific virtual machine and do not wish to create a generalized image,  
# skip the deprovision step  
# sudo rm -rf /var/lib/waagent/  
# sudo rm -f /var/log/waagent.log  
  
# waagent -force -deprovision+user  
# rm -f ~/.bash_history  
  
# export HISTSIZE=0  
  
# logout
```

18. Shut down the virtual machine in KVM.

19. Convert the qcow2 image to the VHD format.

**NOTE**

There is a known bug in qemu-img versions  $\geq 2.2.1$  that results in an improperly formatted VHD. The issue has been fixed in QEMU 2.6. It is recommended to use either qemu-img 2.2.0 or lower, or update to 2.6 or higher. Reference: <https://bugs.launchpad.net/qemu/+bug/1490611>.

First convert the image to raw format:

```
# qemu-img convert -f qcow2 -O raw rhel-6.9.qcow2 rhel-6.9.raw
```

Make sure that the size of the raw image is aligned with 1 MB. Otherwise, round up the size to align with 1 MB:

```
# MB=$((1024*1024))
# size=$(qemu-img info -f raw --output json "rhel-6.9.raw" | \
#       gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')
#
# rounded_size=$(((size/$MB + 1)*$MB))
# qemu-img resize rhel-6.9.raw $rounded_size
```

Convert the raw disk to a fixed-sized VHD:

```
# qemu-img convert -f raw -o subformat=fixed -O vpc rhel-6.9.raw rhel-6.9.vhd
```

Or, with qemu version 2.6+ include the `force_size` option:

```
# qemu-img convert -f raw -o subformat=fixed,force_size -O vpc rhel-6.9.raw rhel-6.9.vhd
```

## RHEL 7 using KVM

1. Download the KVM image of RHEL 7 from the Red Hat website. This procedure uses RHEL 7 as the example.
2. Set a root password.

Generate an encrypted password, and copy the output of the command:

```
# openssl passwd -1 changeme
```

Set a root password with guestfish:

```
# guestfish --rw -a <image-name>
> <fs> run
> <fs> list/filesystems
> <fs> mount /dev/sda1 /
> <fs> vi /etc/shadow
> <fs> exit
```

Change the second field of root user from "!!" to the encrypted password.

3. Create a virtual machine in KVM from the qcow2 image. Set the disk type to **qcow2**, and set the virtual network interface device model to **virtio**. Then, start the virtual machine, and sign in as root.
4. Create or edit the `/etc/sysconfig/network` file, and add the following text:

```
NETWORKING=yes
HOSTNAME=localhost.localdomain
```

5. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
TYPE=Ethernet
USERCTL=no
PEERDNS=yes
IPV6INIT=no
PERSISTENT_DHCLIENT=yes
NM_CONTROLLED=yes
```

6. Ensure that the network service will start at boot time by running the following command:

```
# sudo systemctl enable network
```

7. Register your Red Hat subscription to enable installation of packages from the RHEL repository by running the following command:

```
# subscription-manager register --auto-attach --username=XXX --password=XXX
```

8. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure.

To do this configuration, open `/etc/default/grub` in a text editor, and edit the `GRUB_CMDLINE_LINUX` parameter. For example:

```
GRUB_CMDLINE_LINUX="console=ttyS0 earlyprintk=ttyS0 net.ifnames=0"
```

This command also ensures that all console messages are sent to the first serial port, which can assist Azure support with debugging issues. The command also turns off the new RHEL 7 naming conventions for NICs. In addition, we recommend that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

9. After you are done editing `/etc/default/grub`, run the following command to rebuild the grub configuration:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

10. Add Hyper-V modules into initramfs.

Edit `/etc/dracut.conf` and add content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild initramfs:

```
# dracut -f -v
```

11. Uninstall cloud-init:

```
# yum remove cloud-init
```

12. Ensure that the SSH server is installed and configured to start at boot time:

```
# systemctl enable sshd
```

Modify /etc/ssh/sshd\_config to include the following lines:

```
PasswordAuthentication yes  
ClientAliveInterval 180
```

13. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository.

Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-7-server-extras-rpms
```

14. Install the Azure Linux Agent by running the following command:

```
# yum install WALinuxAgent
```

Enable the waagent service:

```
# systemctl enable waagent.service
```

15. Install cloud-init Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 12, 'Install cloud-init to handle the provisioning.'

16. Swap configuration

Do not create swap space on the operating system disk. Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 13, 'Swap configuration'

17. Unregister the subscription (if necessary) by running the following command:

```
# subscription-manager unregister
```

18. Deprovision

Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 15, 'Deprovision'

19. Shut down the virtual machine in KVM.

20. Convert the qcow2 image to the VHD format.

**NOTE**

There is a known bug in qemu-img versions  $\geq 2.2.1$  that results in an improperly formatted VHD. The issue has been fixed in QEMU 2.6. It is recommended to use either qemu-img 2.2.0 or lower, or update to 2.6 or higher.

Reference: <https://bugs.launchpad.net/qemu/+bug/1490611>.

First convert the image to raw format:

```
# qemu-img convert -f qcow2 -O raw rhel-7.4.qcow2 rhel-7.4.raw
```

Make sure that the size of the raw image is aligned with 1 MB. Otherwise, round up the size to align with 1 MB:

```
# MB=$((1024*1024))
# size=$(qemu-img info -f raw --output json "rhel-7.4.raw" | \
gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')

# rounded_size=$(((size/$MB + 1)*$MB))
# qemu-img resize rhel-7.4.raw $rounded_size
```

Convert the raw disk to a fixed-sized VHD:

```
# qemu-img convert -f raw -o subformat=fixed -O vpc rhel-7.4.raw rhel-7.4.vhd
```

Or, with qemu version 2.6+ include the `force_size` option:

```
# qemu-img convert -f raw -o subformat=fixed,force_size -O vpc rhel-7.4.raw rhel-7.4.vhd
```

## VMware

This section shows you how to prepare a [RHEL 6](#) or [RHEL 7](#) distro from VMware.

### Prerequisites

This section assumes that you have already installed a RHEL virtual machine in VMware. For details about how to install an operating system in VMware, see [VMware Guest Operating System Installation Guide](#).

- When you install the Linux operating system, we recommend that you use standard partitions rather than LVM, which is often the default for many installations. This will avoid LVM name conflicts with cloned virtual machine, particularly if an operating system disk ever needs to be attached to another virtual machine for troubleshooting. LVM or RAID can be used on data disks if preferred.
- Do not configure a swap partition on the operating system disk. You can configure the Linux agent to create a swap file on the temporary resource disk. You can find more information about this in the steps that follow.
- When you create the virtual hard disk, select **Store virtual disk as a single file**.

### RHEL 6 using VMware

1. In RHEL 6, NetworkManager can interfere with the Azure Linux agent. Uninstall this package by running the following command:

```
# sudo rpm -e --nodeps NetworkManager
```

2. Create a file named **network** in the `/etc/sysconfig/` directory that contains the following text:

```
NETWORKING=yes
HOSTNAME=localhost.localdomain
```

3. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
TYPE=Ethernet
USERCTL=no
PEERDNS=yes
IPV6INIT=no
```

4. Move (or remove) the udev rules to avoid generating static rules for the Ethernet interface. These rules cause problems when you clone a virtual machine in Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules
# sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

5. Ensure that the network service will start at boot time by running the following command:

```
# sudo chkconfig network on
```

6. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# sudo subscription-manager register --auto-attach --username=XXX --password=XXX
```

7. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository. Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-6-server-extras-rpms
```

8. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this, open `/etc/default/grub` in a text editor, and edit the `GRUB_CMDLINE_LINUX` parameter. For example:

```
GRUB_CMDLINE_LINUX="console=ttyS0 earlyprintk=ttyS0"
```

This will also ensure that all console messages are sent to the first serial port, which can assist Azure support with debugging issues. In addition, we recommend that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

9. Add Hyper-V modules to initramfs:

Edit `/etc/dracut.conf`, and add the following content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild initramfs:

```
# dracut -f -v
```

10. Ensure that the SSH server is installed and configured to start at boot time, which is usually the default.

Modify `/etc/ssh/sshd_config` to include the following line:

```
ClientAliveInterval 180
```

11. Install the Azure Linux Agent by running the following command:

```
# sudo yum install WALinuxAgent  
# sudo chkconfig waagent on
```

12. Do not create swap space on the operating system disk.

The Azure Linux Agent can automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. Note that the local resource disk is a temporary disk, and it might be emptied if the virtual machine is deprovisioned. After you install the Azure Linux Agent in the previous step, modify the following parameters in `/etc/waagent.conf` appropriately:

```
ResourceDisk.Format=y  
ResourceDisk.Filesystem=ext4  
ResourceDisk.MountPoint=/mnt/resource  
ResourceDisk.EnableSwap=y  
ResourceDisk.SwapSizeMB=2048    ## NOTE: set this to whatever you need it to be.
```

13. Unregister the subscription (if necessary) by running the following command:

```
# sudo subscription-manager unregister
```

14. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# Note: if you are migrating a specific virtual machine and do not wish to create a generalized image,  
# skip the deprovision step  
# sudo rm -rf /var/lib/waagent/  
# sudo rm -f /var/log/waagent.log  
  
# waagent -force -deprovision+user  
# rm -f ~/.bash_history  
  
# export HISTSIZE=0  
  
# logout
```

15. Shut down the virtual machine, and convert the VMDK file to a .vhdx file.

## NOTE

There is a known bug in qemu-img versions  $\geq 2.2.1$  that results in an improperly formatted VHD. The issue has been fixed in QEMU 2.6. It is recommended to use either qemu-img 2.2.0 or lower, or update to 2.6 or higher. Reference: <https://bugs.launchpad.net/qemu/+bug/1490611>.

First convert the image to raw format:

```
# qemu-img convert -f vmdk -O raw rhel-6.9.vmdk rhel-6.9.raw
```

Make sure that the size of the raw image is aligned with 1 MB. Otherwise, round up the size to align with 1 MB:

```
# MB=$((1024*1024))
# size=$(qemu-img info -f raw --output json "rhel-6.9.raw" | \
gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}')

# rounded_size=$(((size/$MB + 1)*$MB))
# qemu-img resize rhel-6.9.raw $rounded_size
```

Convert the raw disk to a fixed-sized VHD:

```
# qemu-img convert -f raw -o subformat=fixed -O vpc rhel-6.9.raw rhel-6.9.vhd
```

Or, with qemu version 2.6+ include the `force_size` option:

```
# qemu-img convert -f raw -o subformat=fixed,force_size -O vpc rhel-6.9.raw rhel-6.9.vhd
```

## RHEL 7 using VMware

1. Create or edit the `/etc/sysconfig/network` file, and add the following text:

```
NETWORKING=yes
HOSTNAME=localhost.localdomain
```

2. Create or edit the `/etc/sysconfig/network-scripts/ifcfg-eth0` file, and add the following text:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
TYPE=Ethernet
USERCTL=no
PEERDNS=yes
IPV6INIT=no
PERSISTENT_DHCLIENT=yes
NM_CONTROLLED=yes
```

3. Ensure that the network service will start at boot time by running the following command:

```
# sudo systemctl enable network
```

4. Register your Red Hat subscription to enable the installation of packages from the RHEL repository by running the following command:

```
# sudo subscription-manager register --auto-attach --username=XXX --password=XXX
```

5. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure.

To do this modification, open `/etc/default/grub` in a text editor, and edit the `GRUB_CMDLINE_LINUX` parameter. For example:

```
GRUB_CMDLINE_LINUX="console=ttyS0 earlyprintk=ttyS0 net.ifnames=0"
```

This configuration also ensures that all console messages are sent to the first serial port, which can assist Azure support with debugging issues. It also turns off the new RHEL 7 naming conventions for NICs. In addition, we recommend that you remove the following parameters:

```
rhgb quiet crashkernel=auto
```

Graphical and quiet boot are not useful in a cloud environment where we want all the logs to be sent to the serial port. You can leave the `crashkernel` option configured if desired. Note that this parameter reduces the amount of available memory in the virtual machine by 128 MB or more, which might be problematic on smaller virtual machine sizes.

6. After you are done editing `/etc/default/grub`, run the following command to rebuild the grub configuration:

```
# sudo grub2-mkconfig -o /boot/grub2/grub.cfg
```

7. Add Hyper-V modules to initramfs.

Edit `/etc/dracut.conf`, add content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild initramfs:

```
# dracut -f -v
```

8. Ensure that the SSH server is installed and configured to start at boot time. This setting is usually the default. Modify `/etc/ssh/sshd_config` to include the following line:

```
ClientAliveInterval 180
```

9. The WALinuxAgent package, `WALinuxAgent-<version>`, has been pushed to the Red Hat extras repository. Enable the extras repository by running the following command:

```
# subscription-manager repos --enable=rhel-7-server-extras-rpms
```

10. Install the Azure Linux Agent by running the following command:

```
# sudo yum install WALinuxAgent  
# sudo systemctl enable waagent.service
```

## 11. Install cloud-init

Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 12, 'Install cloud-init to handle the provisioning.'

## 12. Swap configuration

Do not create swap space on the operating system disk. Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 13, 'Swap configuration'

## 13. If you want to unregister the subscription, run the following command:

```
# sudo subscription-manager unregister
```

## 14. Deprovision

Follow the steps in 'Prepare a RHEL 7 virtual machine from Hyper-V Manager', step 15, 'Deprovision'

## 15. Shut down the virtual machine, and convert the VMDK file to the VHD format.

### NOTE

There is a known bug in qemu-img versions >=2.2.1 that results in an improperly formatted VHD. The issue has been fixed in QEMU 2.6. It is recommended to use either qemu-img 2.2.0 or lower, or update to 2.6 or higher.

Reference: <https://bugs.launchpad.net/qemu/+bug/1490611>.

First convert the image to raw format:

```
# qemu-img convert -f vmdk -O raw rhel-7.4.vmdk rhel-7.4.raw
```

Make sure that the size of the raw image is aligned with 1 MB. Otherwise, round up the size to align with 1 MB:

```
# MB=$((1024*1024))  
# size=$(qemu-img info -f raw --output json "rhel-7.4.raw" | \  
# gawk 'match($0, /"virtual-size": ([0-9]+),/, val) {print val[1]}' )  
  
# rounded_size=$(((size/$MB + 1)*$MB))  
# qemu-img resize rhel-7.4.raw $rounded_size
```

Convert the raw disk to a fixed-sized VHD:

```
# qemu-img convert -f raw -O vpc rhel-7.4.raw rhel-7.4.vhd
```

Or, with qemu version 2.6+ include the `force_size` option:

```
# qemu-img convert -f raw -O vpc,force_size rhel-7.4.raw rhel-7.4.vhd
```

# Kickstart file

This section shows you how to prepare a RHEL 7 distro from an ISO using a kickstart file.

## RHEL 7 from a kickstart file

1. Create a kickstart file that includes the following content, and save the file. For details about kickstart installation, see the [Kickstart Installation Guide](#).

```
# Kickstart for provisioning a RHEL 7 Azure VM

# System authorization information
auth --enableshadow --passalgo=sha512

# Use graphical install
text

# Do not run the Setup Agent on first boot
firstboot --disable

# Keyboard layouts
keyboard --vckeymap=us --xlayouts='us'

# System language
lang en_US.UTF-8

# Network information
network --bootproto=dhcp

# Root password
rootpw --plaintext "to_be_disabled"

# System services
services --enabled="sshd,waagent,NetworkManager"

# System timezone
timezone Etc/UTC --isUtc --ntpservers
0.rhel.pool.ntp.org,1.rhel.pool.ntp.org,2.rhel.pool.ntp.org,3.rhel.pool.ntp.org

# Partition clearing information
clearpart --all --initlabel

# Clear the MBR
zerombr

# Disk partitioning information
part /boot --fstype="xfs" --size=500
part / --fstype="xfs" --size=1 --grow --asprimary

# System bootloader configuration
bootloader --location=mbr

# Firewall configuration
firewall --disabled

# Enable SELinux
selinux --enforcing

# Don't configure X
skipx

# Power down the machine after install
poweroff

%packages
@base
@console-internet
```

```

crony
sudo
parted
-dracut-config-rescue

%end

%post --log=/var/log/anaconda/post-install.log

#!/bin/bash

# Register Red Hat Subscription
subscription-manager register --username=XXX --password=XXX --auto-attach --force

# Install latest repo update
yum update -y

# Enable extras repo
subscription-manager repos --enable=rhel-7-server-extras-rpms

# Install WALinuxAgent
yum install -y WALinuxAgent

# Unregister Red Hat subscription
subscription-manager unregister

# Enable waaagent at boot-up
systemctl enable waagent

# Install cloud-init
yum install -y cloud-init cloud-utils-growpart gdisk hyperv-daemons

# Configure waagent for cloud-init
sed -i 's/Provisioning.Agent=auto/Provisioning.Agent=cloud-init/g' /etc/waagent.conf
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf

echo "Adding mounts and disk_setup to init stage"
sed -i '/ - mounts/d' /etc/cloud/cloud.cfg
sed -i '/ - disk_setup/d' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - mounts' /etc/cloud/cloud.cfg
sed -i '/cloud_init_modules/a\\ - disk_setup' /etc/cloud/cloud.cfg

# Disable the root account
usermod root -p '!!!

# Configure swap using cloud-init
echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
ephemeral0:
  table_type: mbr
  layout: [66, [33, 82]]
  overwrite: True
fs_setup:
- device: ephemeral0.1
  filesystem: ext4
- device: ephemeral0.2
  filesystem: swap
mounts:
- ["ephemeral0.1", "/mnt"]
- ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.device-timeout=2,x-systemd.requires=cloud-init.service", "0", "0"]
EOF

# Set the cmdline

```

```

sed -i 's/^\\(GRUB_CMDLINE_LINUX\\)=.*"$\\1="console=tty1 console=ttyS0 earlyprintk=ttyS0"/g'
/etc/default/grub

# Enable SSH keepalive
sed -i 's/^#\\(ClientAliveInterval\\).*$\\1 180/g' /etc/ssh/sshd_config

# Build the grub cfg
grub2-mkconfig -o /boot/grub2/grub.cfg

# Configure network
cat << EOF > /etc/sysconfig/network-scripts/ifcfg-eth0
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=dhcp
TYPE=Ethernet
USERCTL=no
PEERDNS=yes
IPV6INIT=no
PERSISTENT_DHCLIENT=yes
NM_CONTROLLED=yes
EOF

# Deprovision and prepare for Azure if you are creating a generalized image
sudo cloud-init clean --logs --seed
sudo rm -rf /var/lib/cloud/
sudo rm -rf /var/lib/waagent/
sudo rm -f /var/log/waagent.log

sudo waagent -force -deprovision+user
rm -f ~/.bash_history
export HISTSIZE=0

%end

```

2. Place the kickstart file where the installation system can access it.
3. In Hyper-V Manager, create a new virtual machine. On the **Connect Virtual Hard Disk** page, select **Attach a virtual hard disk later**, and complete the New Virtual Machine Wizard.
4. Open the virtual machine settings:
  - a. Attach a new virtual hard disk to the virtual machine. Make sure to select **VHD Format** and **Fixed Size**.
  - b. Attach the installation ISO to the DVD drive.
  - c. Set the BIOS to boot from CD.
5. Start the virtual machine. When the installation guide appears, press **Tab** to configure the boot options.
6. Enter `inst.ks=<the location of the kickstart file>` at the end of the boot options, and press **Enter**.
7. Wait for the installation to finish. When it's finished, the virtual machine will be shut down automatically. Your Linux VHD is now ready to be uploaded to Azure.

## Known issues

### The Hyper-V driver could not be included in the initial RAM disk when using a non-Hyper-V hypervisor

In some cases, Linux installers might not include the drivers for Hyper-V in the initial RAM disk (initrd or initramfs) unless Linux detects that it is running in a Hyper-V environment.

When you're using a different virtualization system (that is, VirtualBox, Xen, etc.) to prepare your Linux image, you might need to rebuild initrd to ensure that at least the hv\_vmbus and hv\_storvsc kernel modules are available on the initial RAM disk. This is a known issue at least on systems that are based on the upstream Red

Hat distribution.

To resolve this issue, add Hyper-V modules to initramfs and rebuild it:

Edit `/etc/dracut.conf`, and add the following content:

```
add_drivers+=" hv_vmbus hv_netvsc hv_storvsc "
```

Rebuild initramfs:

```
# dracut -f -v
```

For more details, see the information about [rebuilding initramfs](#).

## Next steps

- You're now ready to use your Red Hat Enterprise Linux virtual hard disk to create new virtual machines in Azure. If this is the first time that you're uploading the .vhdx file to Azure, see [Create a Linux VM from a custom disk](#).
- For more details about the hypervisors that are certified to run Red Hat Enterprise Linux, see [the Red Hat website](#).
- To learn more about using production-ready RHEL BYOS images, go to the documentation page for [BYOS](#).

# Prepare a SLES or openSUSE Leap virtual machine for Azure

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets Applies to: ✓ Uniform scale sets

This article assumes that you have already installed a SUSE or openSUSE Leap Linux operating system to a virtual hard disk. Multiple tools exist to create .vhd files, for example a virtualization solution such as Hyper-V. For instructions, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

## SLES / openSUSE Leap installation notes

- Please see also [General Linux Installation Notes](#) for more tips on preparing Linux for Azure.
- The VHDX format is not supported in Azure, only **fixed VHD**. You can convert the disk to VHD format using Hyper-V Manager or the convert-vhd cmdlet.
- When installing the Linux system it is recommended that you use standard partitions rather than LVM (often the default for many installations). This will avoid LVM name conflicts with cloned VMs, particularly if an OS disk ever needs to be attached to another VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks if preferred.
- Do not configure a swap partition on the OS disk. The Linux agent can be configured to create a swap file on the temporary resource disk. More information about this can be found in the steps below.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1MB before conversion. See [Linux Installation Notes](#) for more information.

## Use SUSE Studio

[SUSE Studio](#) can easily create and manage your SLES and openSUSE Leap images for Azure and Hyper-V. This is the recommended approach for customizing your own SLES and openSUSE Leap images.

As an alternative to building your own VHD, SUSE also publishes BYOS (Bring Your Own Subscription) images for SLES at [VM Depot](#).

## Prepare SUSE Linux Enterprise Server for Azure

1. In the center pane of Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open the window for the virtual machine.
3. Register your SUSE Linux Enterprise system to allow it to download updates and install packages.
4. Update the system with the latest patches:

```
# sudo zypper update
```

5. Install Azure Linux Agent and cloud-init

```
# SUSEConnect -p sle-module-public-cloud/15.2/x86_64 (SLES 15 SP2)
# sudo zypper refresh
# sudo zypper install python-azure-agent
# sudo zypper install cloud-init
```

## 6. Enable waagent & cloud-init to start on boot

```
# sudo chkconfig waagent on
# systemctl enable cloud-init-local.service
# systemctl enable cloud-init.service
# systemctl enable cloud-config.service
# systemctl enable cloud-final.service
# systemctl daemon-reload
# cloud-init clean
```

## 7. Update waagent and cloud-init configuration

```
# sed -i 's/Provisioning.UseCloudInit=n/Provisioning.UseCloudInit=y/g' /etc/waagent.conf
# sed -i 's/Provisioning.Enabled=y/Provisioning.Enabled=n/g' /etc/waagent.conf

# sudo sh -c 'printf "datasource:\n Azure:" > /etc/cloud/cloud.cfg.d/91-azure_datasource.cfg'
# sudo sh -c 'printf "reporting:\n logging:\n type: log\n telemetry:\n type: hyperv" >
/etc/cloud/cloud.cfg.d/10-azure-kvp.cfg'
```

## 8. Edit /etc/default/grub file to ensure console logs are sent to serial port and then update the main configuration file with grub2-mkconfig -o /boot/grub2/grub.cfg

```
console=ttyS0 earlyprintk=ttyS0
```

This will ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues.

## 9. Ensure /etc/fstab file reference the disk using its UUID (by-uuid)

## 10. Modify udev rules to avoid generating static rules for the Ethernet interface(s). These rules can cause problems when cloning a virtual machine in Microsoft Azure or Hyper-V:

```
# sudo ln -s /dev/null /etc/udev/rules.d/75-persistent-net-generator.rules
# sudo rm -f /etc/udev/rules.d/70-persistent-net.rules
```

## 11. It is recommended to edit the file "/etc/sysconfig/network/dhcp" and change the `DHCLIENT_SET_HOSTNAME` parameter to the following:

```
DHCLIENT_SET_HOSTNAME="no"
```

## 12. In "/etc/sudoers", comment out or remove the following lines if they exist:

```
Defaults targetpw  # ask for the password of the target user i.e. root
ALL      ALL=(ALL) ALL    # WARNING! Only use this together with 'Defaults targetpw'!
```

## 13. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.

## 14. Swap configuration

Do not create swap space on the operating system disk.

Previously, the Azure Linux Agent was used automatically configure swap space by using the local resource disk that is attached to the virtual machine after the virtual machine is provisioned on Azure. However this is now handled by cloud-init, you **must not** use the Linux Agent to format the resource disk create the swap file, modify the following parameters in `/etc/waagent.conf` appropriately:

```
# sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
# sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf
```

If you want mount, format and create swap you can either:

- Pass this in as a cloud-init config every time you create a VM.
- Use a cloud-init directive baked into the image that will do this every time the VM is created:

```
echo 'DefaultEnvironment="CLOUD_CFG=/etc/cloud/cloud.cfg.d/00-azure-swap.cfg"' >>
/etc/systemd/system.conf
cat > /etc/cloud/cloud.cfg.d/00-azure-swap.cfg << EOF
#cloud-config
# Generated by Azure cloud image build
disk_setup:
ephemeral0:
  table_type: mbr
  layout: [66, [33, 82]]
  overwrite: True
fs_setup:
  - device: ephemeral0.1
    filesystem: ext4
  - device: ephemeral0.2
    filesystem: swap
mounts:
  - ["ephemeral0.1", "/mnt"]
  - ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service,x-
systemd.device-timeout=2", "0", "0"]
EOF
```

15. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# sudo rm -rf /var/lib/waagent/
# sudo rm -f /var/log/waagent.log

# waagent -force -deprovision+user
# rm -f ~/.bash_history

# export HISTSIZE=0

# logout
```

16. Click Action -> **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## Prepare openSUSE 15.2+

1. In the center pane of Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open the window for the virtual machine.

3. On the shell, run the command '`zypper lr`'. If this command returns output similar to the following, then the repositories are configured as expected--no adjustments are necessary (note that version numbers may vary):

| # | ALIAS                 | NAME                  | ENABLED | REFRESH |
|---|-----------------------|-----------------------|---------|---------|
| 1 | Cloud:Tools_15.2      | Cloud:Tools_15.2      | Yes     | Yes     |
| 2 | openSUSE_15.2_OS<br>S | openSUSE_15.2_OS<br>S | Yes     | Yes     |
| 3 | openSUSE_15.2_Updates | openSUSE_15.2_Updates | Yes     | Yes     |

If the command returns "No repositories defined..." then use the following commands to add these repos:

```
# sudo zypper ar -f http://download.opensuse.org/repositories/Cloud:Tools/openSUSE_15.2  
Cloud:Tools_15.2  
# sudo zypper ar -f https://download.opensuse.org/distribution/15.2/repo/oss openSUSE_15.2_OSS  
# sudo zypper ar -f http://download.opensuse.org/update/15.2 openSUSE_15.2_Updates
```

You can then verify the repositories have been added by running the command '`zypper lr`' again. In case one of the relevant update repositories is not enabled, enable it with following command:

```
# sudo zypper mr -e [NUMBER OF REPOSITORY]
```

4. Update the kernel to the latest available version:

```
# sudo zypper up kernel-default
```

Or to update the system with all the latest patches:

```
# sudo zypper update
```

5. Install the Azure Linux Agent.

```
# sudo zypper install WALinuxAgent
```

6. Modify the kernel boot line in your grub configuration to include additional kernel parameters for Azure. To do this, open "/boot/grub/menu.lst" in a text editor and ensure that the default kernel includes the following parameters:

```
console=ttyS0 earlyprintk=ttyS0
```

This will ensure all console messages are sent to the first serial port, which can assist Azure support with debugging issues. In addition, remove the following parameters from the kernel boot line if they exist:

```
libata.atapi_enabled=0 reserve=0x1f0,0x8
```

7. It is recommended to edit the file "/etc/sysconfig/network/dhcp" and change the `DHCLIENT_SET_HOSTNAME`

parameter to the following:

```
DHCLIENT_SET_HOSTNAME="no"
```

8. **Important:** In "/etc/sudoers", comment out or remove the following lines if they exist:

```
Defaults targetpw  # ask for the password of the target user i.e. root
ALL    ALL=(ALL) ALL  # WARNING! Only use this together with 'Defaults targetpw'!
```

9. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.

10. Do not create swap space on the OS disk.

The Azure Linux Agent can automatically configure swap space using the local resource disk that is attached to the VM after provisioning on Azure. Note that the local resource disk is a *temporary* disk, and might be emptied when the VM is deprovisioned. After installing the Azure Linux Agent (see previous step), modify the following parameters in /etc/waagent.conf appropriately:

```
ResourceDisk.Format=y
ResourceDisk.Filesystem=ext4
ResourceDisk.MountPoint=/mnt/resource
ResourceDisk.EnableSwap=y
ResourceDisk.SwapSizeMB=2048    ## NOTE: set this to whatever you need it to be.
```

11. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

```
# sudo waagent -force -deprovision
# export HISTSIZE=0
# logout
```

12. Ensure the Azure Linux Agent runs at startup:

```
# sudo systemctl enable waagent.service
```

13. Click Action -> **Shut Down** in Hyper-V Manager. Your Linux VHD is now ready to be [uploaded to Azure](#).

## Next steps

You're now ready to use your SUSE Linux virtual hard disk to create new virtual machines in Azure. If this is the first time that you're uploading the .vhd file to Azure, see [Create a Linux VM from a custom disk](#).

# Prepare an Ubuntu virtual machine for Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Ubuntu now publishes official Azure VHDs for download at <https://cloud-images.ubuntu.com/>. If you need to build your own specialized Ubuntu image for Azure, rather than use the manual procedure below it is recommended to start with these known working VHDs and customize as needed. The latest image releases can always be found at the following locations:

- Ubuntu 18.04/Bionic: [bionic-server-cloudimg-amd64-azure.vhd.zip](#)
- Ubuntu 20.04/Focal: [focal-server-cloudimg-amd64-azure.vhd.zip](#)

## Prerequisites

This article assumes that you have already installed an Ubuntu Linux operating system to a virtual hard disk. Multiple tools exist to create .vhd files, for example a virtualization solution such as Hyper-V. For instructions, see [Install the Hyper-V Role and Configure a Virtual Machine](#).

### Ubuntu installation notes

- Please see also [General Linux Installation Notes](#) for more tips on preparing Linux for Azure.
- The VHDX format is not supported in Azure, only **fixed VHD**. You can convert the disk to VHD format using Hyper-V Manager or the `Convert-VHD` cmdlet.
- When installing the Linux system it is recommended that you use standard partitions rather than LVM (often the default for many installations). This will avoid LVM name conflicts with cloned VMs, particularly if an OS disk ever needs to be attached to another VM for troubleshooting. [LVM](#) or [RAID](#) may be used on data disks if preferred.
- Do not configure a swap partition or swapfile on the OS disk. The cloud-init provisioning agent can be configured to create a swap file or a swap partition on the temporary resource disk. More information about this can be found in the steps below.
- All VHDs on Azure must have a virtual size aligned to 1MB. When converting from a raw disk to VHD you must ensure that the raw disk size is a multiple of 1MB before conversion. See [Linux Installation Notes](#) for more information.

## Manual steps

### NOTE

Before attempting to create your own custom Ubuntu image for Azure, please consider using the pre-built and tested images from <https://cloud-images.ubuntu.com/> instead.

1. In the center pane of Hyper-V Manager, select the virtual machine.
2. Click **Connect** to open the window for the virtual machine.
3. Replace the current repositories in the image to use Ubuntu's Azure repository.

Before editing `/etc/apt/sources.list`, it is recommended to make a backup:

```
# sudo cp /etc/apt/sources.list /etc/apt/sources.list.bak
```

Ubuntu 18.04 and Ubuntu 20.04:

```
# sudo sed -i  
's/http://archive.ubuntu.com/ubuntu//http://azure.archive.ubuntu.com/ubuntu//g'  
/etc/apt/sources.list  
# sudo sed -i 's/http:///[a-z][a-  
z]archive.ubuntu.com/ubuntu//http://azure.archive.ubuntu.com/ubuntu//g'  
/etc/apt/sources.list  
# sudo apt-get update
```

4. The Ubuntu Azure images are now using the [Azure-tailored kernel](#). Update the operating system to the latest Azure-tailored kernel and install Azure Linux tools (including Hyper-V dependencies) by running the following commands:

Ubuntu 18.04 and Ubuntu 20.04:

```
# sudo apt update  
# sudo apt install linux-azure linux-image-azure linux-headers-azure linux-tools-common linux-cloud-  
tools-common linux-tools-azure linux-cloud-tools-azure  
(recommended) # sudo apt full-upgrade  
  
# sudo reboot
```

5. Modify the kernel boot line for Grub to include additional kernel parameters for Azure. To do this open `/etc/default/grub` in a text editor, find the variable called `GRUB_CMDLINE_LINUX_DEFAULT` (or add it if needed) and edit it to include the following parameters:

```
GRUB_CMDLINE_LINUX_DEFAULT="console=tty1 console=ttyS0,115200n8 earlyprintk=ttyS0,115200  
rootdelay=300 quiet splash"
```

Save and close this file, and then run `sudo update-grub`. This will ensure all console messages are sent to the first serial port, which can assist Azure technical support with debugging issues.

6. Ensure that the SSH server is installed and configured to start at boot time. This is usually the default.
7. Install cloud-init (the provisioning agent) and the Azure Linux Agent (the guest extensions handler). Cloud-init uses `netplan` to configure the system network configuration (during provisioning and each subsequent boot) and `gdisk` to partition resource disks.

```
# sudo apt update  
# sudo apt install cloud-init gdisk netplan.io walinuagent && systemctl stop walinuagent
```

#### NOTE

The `walinuagent` package may remove the `NetworkManager` and `NetworkManager-gnome` packages, if they are installed.

8. Remove cloud-init default configs and leftover `netplan` artifacts that may conflict with cloud-init provisioning on Azure:

```
# rm -f /etc/cloud/cloud.cfg.d/50-curtin-networking.cfg /etc/cloud/cloud.cfg.d/curtin-preserve-sources.cfg /etc/cloud/cloud.cfg.d/99-installer.cfg /etc/cloud/cloud.cfg.d/subiquity-disable-cloudinit-networking.cfg
# rm -f /etc/cloud/ds-identify.cfg
# rm -f /etc/netplan/*.yaml
```

9. Configure cloud-init to provision the system using the Azure datasource:

```
# cat > /etc/cloud/cloud.cfg.d/90_dpkg.cfg << EOF
datasource_list: [ Azure ]
EOF

# cat > /etc/cloud/cloud.cfg.d/90-azure.cfg << EOF
system_info:
  package_mirrors:
    - arches: [i386, amd64]
      failsafe:
        primary: http://archive.ubuntu.com/ubuntu
        security: http://security.ubuntu.com/ubuntu
      search:
        primary:
          - http://azure.archive.ubuntu.com/ubuntu/
        security: []
    - arches: [armhf, armel, default]
      failsafe:
        primary: http://ports.ubuntu.com/ubuntu-ports
        security: http://ports.ubuntu.com/ubuntu-ports
EOF

# cat > /etc/cloud/cloud.cfg.d/10-azure-kvp.cfg << EOF
reporting:
  logging:
    type: log
  telemetry:
    type: hyperv
EOF
```

10. Configure the Azure Linux agent to rely on cloud-init to perform provisioning. Have a look at the [WALinuxAgent project](#) for more information on these options.

```
sed -i 's/Provisioning.Enabled=y/Provisioning.Enabled=n/g' /etc/waagent.conf
sed -i 's/Provisioning.UseCloudInit=n/Provisioning.UseCloudInit=y/g' /etc/waagent.conf
sed -i 's/ResourceDisk.Format=y/ResourceDisk.Format=n/g' /etc/waagent.conf
sed -i 's/ResourceDisk.EnableSwap=y/ResourceDisk.EnableSwap=n/g' /etc/waagent.conf

cat >> /etc/waagent.conf << EOF
# For Azure Linux agent version >= 2.2.45, this is the option to configure,
# enable, or disable the provisioning behavior of the Linux agent.
# Accepted values are auto (default), waagent, cloud-init, or disabled.
# A value of auto means that the agent will rely on cloud-init to handle
# provisioning if it is installed and enabled, which in this case it will.
Provisioning.Agent=auto
EOF
```

11. Clean cloud-init and Azure Linux agent runtime artifacts and logs:

```
# sudo cloud-init clean --logs --seed
# sudo rm -rf /var/lib/cloud/
# sudo systemctl stop walinuxagent.service
# sudo rm -rf /var/lib/waagent/
# sudo rm -f /var/log/waagent.log
```

12. Run the following commands to deprovision the virtual machine and prepare it for provisioning on Azure:

**NOTE**

The `sudo waagent -force -deprovision+user` command generalizes the image by attempting to clean the system and make it suitable for re-provisioning. The `+user` option deletes the last provisioned user account and associated data.

**WARNING**

Deprovisioning using the command above does not guarantee that the image is cleared of all sensitive information and is suitable for redistribution.

```
# sudo waagent -force -deprovision+user  
# rm -f ~/.bash_history  
# export HISTSIZE=0  
# logout
```

13. Click Action -> Shut Down in Hyper-V Manager.

14. Azure only accepts fixed-size VHDs. If the VM's OS disk is not a fixed-size VHD, use the `Convert-VHD` PowerShell cmdlet and specify the `-VHDTtype Fixed` option. Please have a look at the docs for `Convert-VHD` here: [Convert-VHD](#).

15. To bring a Generation 2 VM on Azure, follow these steps:

- a. Change directory to the boot EFI directory:

```
# cd /boot/efi/EFI
```

- b. Copy the ubuntu directory to a new directory named boot:

```
# sudo cp -r ubuntu/ boot
```

- c. Change directory to the newly created boot directory:

```
# cd boot
```

- d. Rename the shimx64.efi file:

```
# sudo mv shimx64.efi bootx64.efi
```

- e. Rename the grub.cfg file to bootx64.cfg:

```
# sudo mv grub.cfg bootx64.cfg
```

## Next steps

You're now ready to use your Ubuntu Linux virtual hard disk to create new virtual machines in Azure. If this is

the first time that you're uploading the .vhdx file to Azure, see [Create a Linux VM from a custom disk](#).

# cloud-init support for virtual machines in Azure

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article explains the support that exists for [cloud-init](#) to configure a virtual machine (VM) or virtual machine scale sets at provisioning time in Azure. These cloud-init configurations are run on first boot once the resources have been provisioned by Azure.

VM Provisioning is the process where the Azure will pass down your VM Create parameter values, such as hostname, username, password etc., and make them available to the VM as it boots up. A 'provisioning agent' will consume those values, configure the VM, and report back when completed.

Azure supports two provisioning agents [cloud-init](#), and the [Azure Linux Agent \(WALA\)](#).

## cloud-init overview

[cloud-init](#) is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files, or to configure users and security. Because cloud-init is called during the initial boot process, there are no additional steps or required agents to apply your configuration. For more information on how to properly format your `#cloud-config` files or other inputs, see the [cloud-init documentation site](#). `#cloud-config` files are text files encoded in base64.

cloud-init also works across distributions. For example, you don't use `apt-get install` or `yum install` to install a package. Instead you can define a list of packages to install. cloud-init automatically uses the native package management tool for the distro you select.

We are actively working with our endorsed Linux distro partners in order to have cloud-init enabled images available in the Azure Marketplace. These images will make your cloud-init deployments and configurations work seamlessly with VMs and virtual machine scale sets. Initially we collaborate with the endorsed Linux distro partners and upstream to ensure cloud-init functions with the OS on Azure, then the packages are updated and made publicly available in the distro package repositories.

There are two stages to making cloud-init available to the endorsed Linux distro OS's on Azure, package support, and then image support:

- 'cloud-init package support on Azure' documents which cloud-init packages onwards are supported or in preview, so you can use these packages with the OS in a custom image.
- 'image cloud-init ready' documents if the image is already configured to use cloud-init.

### Canonical

| PUBLISHER / VERSION | OFFER        | SKU       | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|--------------|-----------|---------|------------------------|-------------------------------------|
| Canonical 20.04     | UbuntuServer | 20.04-LTS | latest  | yes                    | yes                                 |
| Canonical 18.04     | UbuntuServer | 18.04-LTS | latest  | yes                    | yes                                 |

### RHEL

| PUBLISHER / VERSION | OFFER | SKU                | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|-------|--------------------|---------|------------------------|-------------------------------------|
| RedHat 7            | RHEL  | 7.7, 7.8, 7_9      | latest  | yes                    | yes                                 |
| RedHat 8            | RHEL  | 8.1, 8.2, 8_3, 8_4 | latest  | yes                    | yes                                 |

- All other RedHat SKUs starting from RHEL 7 (version 7.7) and RHEL 8 (version 8.1) including both Gen1 and Gen2 images are provisioned using cloud-init. RHEL 6 images do not support cloud-init.

## CentOS

| PUBLISHER / VERSION | OFFER  | SKU           | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|--------|---------------|---------|------------------------|-------------------------------------|
| OpenLogic 7         | CentOS | 7.7, 7.8, 7.9 | latest  | yes                    | yes                                 |
| OpenLogic 8         | CentOS | 8.1, 8.2, 8.3 | latest  | yes                    | yes                                 |

- All other CentOS SKUs starting from CentOS 7 (version 7.7) and CentOS 8 (version 8.1) including both Gen1 and Gen2 images are provisioned using cloud-init. CentOS 6.10, 7.4, 7.5, and 7.6 images do not support cloud-init.

### NOTE

OpenLogic is now Rogue Wave Software

## Oracle

| PUBLISHER / VERSION | OFFER        | SKU                          | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|--------------|------------------------------|---------|------------------------|-------------------------------------|
| Oracle 7            | Oracle Linux | 77, 78, ol79                 | latest  | yes                    | yes                                 |
| Oracle 8            | Oracle Linux | 81, ol82, ol83-lvm, ol84-lvm | latest  | yes                    | yes                                 |

- All other Oracle SKUs starting from Oracle 7 (version 7.7) and Oracle 8 (version 8.1) including both Gen1 and Gen2 images are provisioned using cloud-init.

## SUSE SLES

| PUBLISHER / VERSION | OFFER                               | SKU           | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|-------------------------------------|---------------|---------|------------------------|-------------------------------------|
| SUSE 15             | SLES (SUSE Linux Enterprise Server) | sp1, sp2, sp3 | latest  | yes                    | yes                                 |

| PUBLISHER / VERSION | OFFER                               | SKU | VERSION | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE |
|---------------------|-------------------------------------|-----|---------|------------------------|-------------------------------------|
| SUSE 12             | SLES (SUSE Linux Enterprise Server) | sp5 | latest  | yes                    | yes                                 |

- All other SUSE SKUs starting from SLES 15 (sp1) and SLES 12 (sp5) including both Gen1 and Gen2 images are provisioned using cloud-init.
- Additionally these images are also provisioned with cloud-init -

| PUBLISHER / VERSION | OFFER                               | SKU / VERSION                              |
|---------------------|-------------------------------------|--------------------------------------------|
| SUSE 12             | SLES (SUSE Linux Enterprise Server) | sles-{byos/sap/sap-byos}:12-sp4:2020.06.10 |
| SUSE 12             | SLES (SUSE Linux Enterprise Server) | sles-{byos/sap/sap-byos}:12-sp3:2020.06.10 |
| SUSE 12             | SLES (SUSE Linux Enterprise Server) | sles-{byos/sap/sap-byos}:12-sp2:2020.06.10 |
| SUSE 15             | SLES (SUSE Linux Enterprise Server) | manager-proxy-4-byosgen1:2020.06.10        |
| SUSE 15             | SLES (SUSE Linux Enterprise Server) | manager-server-4-byos:gen1:2020.06.10      |

## Debian

| PUBLISHER / VERSION | OFFER     | SKU               | VERSION           | IMAGE CLOUD-INIT READY | CLOUD-INIT PACKAGE SUPPORT ON AZURE                   |
|---------------------|-----------|-------------------|-------------------|------------------------|-------------------------------------------------------|
| debian (Gen1)       | debian-10 | 10-cloudinit      | 10:0.20201013.422 | yes                    | yes - support from package version:<br>20.2-2~deb10u1 |
| debian (Gen2)       | debian-10 | 10-cloudinit-gen2 | 0.20201013.422    | yes                    | yes - support from package version:<br>20.2-2~deb10u1 |

Currently Azure Stack will support the provisioning of cloud-init enabled images.

## What is the difference between cloud-init and the Linux Agent (WALA)?

WALA is an Azure platform-specific agent used to provision and configure VMs, and handle [Azure extensions](#).

We are enhancing the task of configuring VMs to use cloud-init instead of the Linux Agent in order to allow existing cloud-init customers to use their current cloud-init scripts, or new customers to take advantage of the

rich cloud-init configuration functionality. If you have existing investments in cloud-init scripts for configuring Linux systems, there are **no additional settings required** to enable cloud-init process them.

cloud-init cannot process Azure extensions, so WALA is still required in the image to process extensions, but will need to have its provisioning code disabled, for endorsed Linux distros images that are being converted to provision by cloud-init, they will have WALA installed, and setup correctly.

When creating a VM, if you do not include the Azure CLI `--custom-data` switch at provisioning time, cloud-init or WALA takes the minimal VM provisioning parameters required to provision the VM and complete the deployment with the defaults. If you reference the cloud-init configuration with the `--custom-data` switch, whatever is contained in your custom data will be available to cloud-init when the VM boots.

cloud-init configurations applied to VMs do not have time constraints and will not cause a deployment to fail by timing out. This is not true for WALA, if you change the WALA defaults to process custom-data, it cannot exceed the total VM provisioning time allowance of 40mins, if so, the VM Create will fail.

## cloud-init VM provisioning without a UDF driver

Beginning with cloud-init 21.2, you can use cloud-init to provision a VM in Azure without a UDF driver. If a UDF driver isn't available in the image, cloud-init uses the metadata that's available in the Azure Instance Metadata Service to provision the VM. Note that this option works only for SSH key and [user data](#). To pass in a password or custom data to a VM during provisioning, you must use a UDF driver.

## Deploying a cloud-init enabled Virtual Machine

Deploying a cloud-init enabled virtual machine is as simple as referencing a cloud-init enabled distribution during deployment. Linux distribution maintainers have to choose to enable and integrate cloud-init into their base Azure published images. Once you have confirmed the image you want to deploy is cloud-init enabled, you can use the Azure CLI to deploy the image.

The first step in deploying this image is to create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed.

The following example creates a resource group named *myResourceGroup* in the *eastus* location.

```
az group create --name myResourceGroup --location eastus
```

The next step is to create a file in your current shell, named *cloud-init.txt* and paste the following configuration. For this example, create the file in the Cloud Shell not on your local machine. You can use any editor you wish. Enter `sensible-editor cloud-init.txt` to create the file and see a list of available editors. Choose #1 to use the `nano` editor. Make sure that the whole cloud-init file is copied correctly, especially the first line:

```
#cloud-config
package_upgrade: true
packages:
- httpd
```

### NOTE

cloud-init has multiple [input types](#), cloud-init will use first line of the customData/userData to indicate how it should process the input, for example `#cloud-config` indicates that the content should be processed as a cloud-init config.

Press `Ctrl + X` to exit the file, type `y` to save the file, and press `Enter` to confirm the file name on exit.

The final step is to create a VM with the [az vm create](#) command.

The following example creates a VM named `centos74` and creates SSH keys if they do not already exist in a default key location. To use a specific set of keys, use the `--ssh-key-value` option. Use the `--custom-data` parameter to pass in your cloud-init config file. Provide the full path to the `cloud-init.txt` config if you saved the file outside of your present working directory.

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS-CI:7-CI:latest \
--custom-data cloud-init.txt \
--generate-ssh-keys
```

When the VM has been created, the Azure CLI shows information specific to your deployment. Take note of the `publicIpAddress`. This address is used to access the VM. It takes some time for the VM to be created, the packages to install, and the app to start. There are background tasks that continue to run after the Azure CLI returns you to the prompt. You can SSH into the VM and use the steps outlined in the Troubleshooting section to view the cloud-init logs.

You can also deploy a cloud-init enabled VM by passing the [parameters in ARM template](#).

## Troubleshooting cloud-init

Once the VM has been provisioned, cloud-init will run through all the modules and script defined in `--custom-data` in order to configure the VM. If you need to troubleshoot any errors or omissions from the configuration, you need to search for the module name (`disk_setup` or `runcmd` for example) in the cloud-init log - located in `/var/log/cloud-init.log`.

### NOTE

Not every module failure results in a fatal cloud-init overall configuration failure. For example, using the `runcmd` module, if the script fails, cloud-init will still report provisioning succeeded because the runcmd module executed.

For more details of cloud-init logging, refer to the [cloud-init documentation](#)

## Telemetry

cloud-init collects usage data and sends it to Microsoft to help improve our products and services. Telemetry is only collected during the provisioning process (first boot of the VM). The data collected helps us investigate provisioning failures and monitor performance and reliability. Data collected does not include any personally identifiable information. Read our [privacy statement](#) to learn more. Some examples of telemetry being collected are (this is not an exhaustive list): OS-related information (cloud-init version, distro version, kernel version), performance metrics of essential VM provisioning actions (time to obtain DHCP lease, time to retrieve metadata necessary to configure the VM, etc.), cloud-init log, and dmesg log.

Telemetry collection is currently enabled for a majority of our marketplace images that use cloud-init. It is enabled by specifying KVP telemetry reporter for cloud-init. In a majority of Azure marketplace images, this configuration can be found in the file `/etc/cloud/cloud.cfg.d/10-azure-kvp.cfg`. Removing this file during image preparation will disable telemetry collection for any VM created from this image.

Sample content of `10-azure-kvp.cfg`

```
reporting:  
  logging:  
    type: log  
  telemetry:  
    type: hyperv
```

## Next steps

[Troubleshoot issues with cloud-init.](#)

For cloud-init examples of configuration changes, see the following documents:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Diving deeper into cloud-init

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

To learn more about [cloud-init](#) or troubleshoot it at a deeper level, you need to understand how it works. This document highlights the important parts, and explains the Azure specifics.

When cloud-init is included in a generalized image, and a VM is created from that image, it will process configurations and run through 5 stages during the initial boot. These stages matter, as it shows you at what point cloud-init will apply configurations.

## Understand Cloud-Init configuration

Configuring a VM to run on a platform, means cloud-init needs to apply multiple configurations, as an image consumer, the main configurations you will be interacting with is `user data` (customData), which supports multiple formats. For more information, see [User-Data Formats & cloud-init 21.2 documentation](#). You also have the ability to add and run scripts (`/var/lib/cloud/scripts`) for additional configuration, below discusses this in more detail.

Some configurations are already baked into Azure Marketplace images that come with cloud-init, such as:

1. **Cloud data source** - cloud-init contains code that can interact with cloud platforms, these are called 'datasources'. When a VM is created from a cloud-init image in [Azure](#), cloud-init loads the Azure datasource, which will interact with the Azure metadata endpoints to get the VM specific configuration.
2. **Runtime config** (`/run/cloud-init`)
3. **Image config** (`/etc/cloud`), like `/etc/cloud/cloud.cfg`, `/etc/cloud/cloud.cfg.d/*.cfg`. An example of where this is used in Azure, it is common for the Linux OS images with cloud-init to have an Azure datasource directive, that tells cloud-init what datasource it should use, this saves cloud-init time:

```
/etc/cloud/cloud.cfg.d# cat 90_dpkg.cfg
# to update this file, run dpkg-reconfigure cloud-init
datasource_list: [ Azure ]
```

## Cloud-init boot stages (processing configuration)

When provisioning with cloud-init, there are 5 stages of boot, which process configuration, and shown in the logs.

1. **Generator Stage:** The cloud-init systemd generator starts, and determines that cloud-init should be included in the boot goals, and if so, it enables cloud-init.
2. **Cloud-init Local Stage:** Here cloud-init will look for the local "Azure" datasource, which will enable cloud-init to interface with Azure, and apply a networking configuration, including fallback.
3. **Cloud-init init Stage (Network):** Networking should be online, and the NIC and route table information should be generated. At this stage, the modules listed in `cloud_init_modules` in `/etc/cloud/cloud.cfg` will be run. The VM in Azure will be mounted, the ephemeral disk is formatted, the hostname is set, along with other tasks.

These are some of the `cloud_init_modules`:

```
- migrator
- seed_random
- bootcmd
- write-files
- growpart
- resizefs
- disk_setup
- mounts
- set_hostname
- update_hostname
- ssh
```

After this stage, cloud-init will signal to the Azure platform that the VM has been provisioned successfully. Some modules may have failed, not all module failures will result in a provisioning failure.

4. **Cloud-init Config Stage:** At this stage, the modules in `cloud_config_modules` defined and listed in `/etc/cloud/cloud.cfg` will be run.
5. **Cloud-init Final Stage:** At this final stage, the modules in `cloud_final_modules`, listed in `/etc/cloud/cloud.cfg`, will be run. Here modules that need to be run late in the boot process run, such as package installations and run scripts etc.
  - During this stage, you can run scripts by placing them in the directories under `/var/lib/cloud/scripts` :
    - `per-boot` - scripts within this directory, run on every reboot
    - `per-instance` - scripts within this directory run when a new instance is first booted
    - `per-once` - scripts within this directory run only once

## Next steps

[Troubleshooting cloud-init](#).

# Troubleshooting VM provisioning with cloud-init

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

If you have been creating generalized custom images, using cloud-init to do provisioning, but have found that VM did not create correctly, you will need to troubleshoot your custom images.

Some examples, of issues with provisioning:

- VM gets stuck at 'creating' for 40 minutes, and the VM creation is marked as failed
- `CustomData` does not get processed
- The ephemeral disk fails to mount
- Users do not get created, or there are user access issues
- Networking is not set up correctly
- Swap file or partition failures

This article steps you through how to troubleshoot cloud-init. For more in-depth details, see [cloud-init deep dive](#).

## Step 1: Test the deployment without `customData`

Cloud-init can accept `customData`, that is passed to it, when the VM is created. First you should ensure this is not causing any issues with deployments. Try to provisioning the VM without passing in any configuration. If you find the VM fails to provision, continue with the steps below, if you find the configuration you are passing is not being applied go [step 4](#).

## Step 2: Review image requirements

The primary cause of VM provisioning failure is the OS image doesn't satisfy the prerequisites for running on Azure. Make sure your images are properly prepared before attempting to provision them in Azure.

The following articles illustrate the steps to prepare various linux distributions that are supported in Azure:

- [CentOS-based Distributions](#)
- [Debian Linux](#)
- [Flatcar Container Linux](#)
- [Oracle Linux](#)
- [Red Hat Enterprise Linux](#)
- [SLES & openSUSE](#)
- [Ubuntu](#)
- [Others: Non-Endorsed Distributions](#)

For the [supported Azure cloud-init images](#), the Linux distributions already have all the required packages and configurations in place to correctly provision the image in Azure. If you find your VM is failing to create from your own curated image, try a supported Azure Marketplace image that already is configured for cloud-init, with your optional `customData`. If the `customData` works correctly with an Azure Marketplace image, then there is probably an issue with your curated image.

## Step 3: Collect & review VM logs

When the VM fails to provision, Azure will show 'creating' status, for 20 minutes, and then reboot the VM, and wait another 20 minutes before finally marking the VM deployment as failed, before finally marking it with an `OSProvisioningTimedOut` error.

While the VM is running, you will need the logs from the VM to understand why provisioning failed. To understand why VM provisioning failed, do not stop the VM. Keep the VM running. You will need to keep the failed VM in a running state in order to collect logs. To collect the logs, use one of the following methods:

- [Serial Console](#)
- [Enable Boot Diagnostics](#) before creating the VM and then [View](#) them during the boot.
- [Run AZ VM Repair](#) to attach and mount the OS disk, which will allow you to collect these logs:

```
/var/log/cloud-init*
/var/log/waagent*
/var/log/syslog*
/var/log/rsyslog*
/var/log/messages*
/var/log/kern*
/var/log/dmesg*
/var/log/boot*
```

To start initial troubleshooting, start with the cloud-init logs, and understand where the failure occurred, then use the other logs to deep dive, and provide additional insights.

- `/var/log/cloud-init.log`
- `/var/log/cloud-init-output.log`
- Serial/boot logs

In all logs, start searching for "Failed", "WARNING", "WARN", "err", "error", "ERROR". Setting configuration to ignore case-sensitive searches is recommended.

### TIP

If you are troubleshooting a custom image, you should consider adding a user during the image. If the provisioning fails to set the admin user, you can still log in to the OS.

## Analyzing the logs

Here are more details about what to look for in each cloud-init log.

### `/var/log/cloud-init.log`

By default, all cloud-init events with a priority of debug or higher, are written to `/var/log/cloud-init.log`. This provides verbose logs of every event that occurred during cloud-init initialization.

For example:

```
2019-10-10 04:51:25,321 - util.py[DEBUG]: Failed mount of '/dev/sr0' as 'auto': Unexpected error while running command.
Command: ['mount', '-o', 'ro,sync', '-t', 'auto', u'/dev/sr0', '/run/cloud-init/tmp/tmpLirklc']
Exit code: 32
Reason: -
Stdout:
Stderr: mount: unknown filesystem type 'udf'
2020-01-31 00:21:53,352 - DataSourceAzure.py[WARNING]: /dev/sr0 was not mountable
```

Once you have found an error or warning, read backwards in the cloud-init log to understand what cloud-init was attempting before it hit the error or warning. In many cases cloud-init will have run OS commands or performed provisioning operations prior to the error, which can provide insights as to why errors appeared in the logs. The following example shows that cloud-init attempted to mount a device right before it hit an error.

```
2019-10-10 04:51:24,010 - util.py[DEBUG]: Running command ['mount', '-o', 'ro,sync', '-t', 'auto', u'/dev/sr0', '/run/cloud-init/tmp/tmpXXXXX'] with allowed return codes [0] (shell=False, capture=True)
```

If you have access to the [Serial Console](#), you can try to rerun the command that cloud-init was trying to run.

The logging for `/var/log/cloud-init.log` can also be reconfigured within `/etc/cloud/cloud.cfg.d/05_logging.cfg`. For more details of cloud-init logging, refer to the [cloud-init documentation](#).

### /var/log/cloud-init-output.log

You can get information from the `stdout` and `stderr` during the [stages of cloud-init](#). This normally involves routing table information, networking information, ssh host key verification information, `stdout` and `stderr` for each stage of cloud-init, along with the timestamp for each stage. If desired, `stderr` and `stdout` logging can be reconfigured from `/etc/cloud/cloud.cfg.d/05_logging.cfg`.

### Serial/boot logs

Cloud-init has multiple dependencies, these are documented in required prerequisites for images on Azure, such as networking, storage, ability to mount an ISO, and mount and format the temporary disk. Any of these may throw errors and cause cloud-init to fail. For example, if the VM cannot get a DHCP lease, cloud-init will fail.

If you still cannot isolate why cloud-init failed to provision then you need to understand what cloud-init stages, and when modules run. See [Diving deeper into cloud-init](#) for more details.

## Step 4: Investigate why the configuration isn't being applied

Not every failure in cloud-init results in a fatal provisioning failure. For example, if you are using the `runcmd` module in a cloud-init config, a non-zero exit code from the command it is running will cause the VM provisioning to fail. This is because it runs after core provisioning functionality that happens in the first 3 stages of cloud-init. To troubleshoot why the configuration did not apply, review the logs in Step 3, and cloud-init modules manually. For example:

- `runcmd` - do the scripts run without errors? Run the configuration manually from the terminal to ensure they run as expected.
- Installing packages - does the VM have access to package repositories?
- You should also check the `customData` data configuration that was provided to the VM, this is located in `/var/lib/cloud/instances/<unique-instance-identifier>/user-data.txt`.

## Next steps

If you still cannot isolate why cloud-init did not run the configuration, you need to look more closely at what happens in each cloud-init stage, and when modules run. See [Diving deeper into cloud-init configuration](#) for

more information.

# Use cloud-init to set hostname for a Linux VM in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use [cloud-init](#) to configure a specific hostname on a virtual machine (VM) or virtual machine scale sets (VMSS) at provisioning time in Azure. These cloud-init scripts run on first boot once the resources have been provisioned by Azure. For more information about how cloud-init works natively in Azure and the supported Linux distros, see [cloud-init overview](#)

## Set the hostname with cloud-init

By default, the hostname is the same as the VM name when you create a new virtual machine in Azure. To run a cloud-init script to change this default hostname when you create a VM in Azure with [az vm create](#), specify the cloud-init file with the `--custom-data` switch.

To see upgrade process in action, create a file in your current shell named `cloud_init_hostname.txt` and paste the following configuration. For this example, create the file in the Cloud Shell not on your local machine. You can use any editor you wish. Enter `sensible-editor cloud_init_hostname.txt` to create the file and see a list of available editors. Choose #1 to use the `nano` editor. Make sure that the whole cloud-init file is copied correctly, especially the first line.

```
#cloud-config
fqdn: myhostname
```

Before deploying this image, you need to create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named `myResourceGroup` in the `eastus` location.

```
az group create --name myResourceGroup --location eastus
```

Now, create a VM with [az vm create](#) and specify the cloud-init file with `--custom-data cloud_init_hostname.txt` as follows:

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS:7-CI:latest \
--custom-data cloud_init_hostname.txt \
--generate-ssh-keys
```

Once created, the Azure CLI shows information about the VM. Use the `publicIpAddress` to SSH to your VM. Enter your own address as follows:

```
ssh <publicIpAddress>
```

To see the VM name, use the `hostname` command as follows:

```
hostname
```

The VM should report the hostname as that value set in the cloud-init file, as shown in the following example output:

```
myhostname
```

## Next steps

For additional cloud-init examples of configuration changes, see the following:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Use cloud-init to update and install packages in a Linux VM in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use [cloud-init](#) to update packages on a Linux virtual machine (VM) or virtual machine scale sets at provisioning time in Azure. These cloud-init scripts run on first boot once the resources have been provisioned by Azure. For more information about how cloud-init works natively in Azure and the supported Linux distros, see [cloud-init overview](#)

## Update a VM with cloud-init

For security purposes, you may want to configure a VM to apply the latest updates on first boot. As cloud-init works across different Linux distros, there is no need to specify `apt` or `yum` for the package manager. Instead, you define `package_upgrade` and let the cloud-init process determine the appropriate mechanism for the distro in use.

For this example, we will be using the Azure Cloud Shell. To see the upgrade process in action, create a file named `cloud_init_upgrade.txt` and paste the following configuration.

Select the Try it button on the code block below to open the Cloud Shell. To create the file and see a list of available editors in the Cloud Shell, type the following:

```
sensible-editor cloud_init_upgrade.txt
```

Copy the text below and paste it into the `cloud_init_upgrade.txt` file. Make sure that the whole cloud-init file is copied correctly, especially the first line.

```
#cloud-config
package_upgrade: true
packages:
- httpd
```

Before deploying, you need to create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named `myResourceGroup` in the `eastus` location.

```
az group create --name myCentOSGroup --location eastus
```

Now, create a VM with [az vm create](#) and specify the cloud-init file with the `--custom-data` parameter as follows:

```
az vm create \
--resource-group myCentOSGroup \
--name centos83 \
--image OpenLogic:CentOS:8_3:latest \
--custom-data cloud_init_upgrade.txt \
--admin-username azureuser \
--generate-ssh-keys
```

SSH to the public IP address of your VM shown in the output from the preceding command. Enter your own `publicIpAddress` as follows:

```
ssh azureuser@<publicIpAddress>
```

Run the package management tool and check for updates.

```
sudo yum update
```

As cloud-init checked for and installed updates on boot, there should be no additional updates to apply. You see the update process, number of altered packages as well as the installation of `httpd` by running `yum history` and review the output similar to the one below.

| ID | Command line     | Date and time    | Action(s) | Altered |
|----|------------------|------------------|-----------|---------|
| 3  | -y install httpd | 2022-02-18 18:30 | Install   | 7       |
| 2  | -y upgrade       | 2022-02-18 18:23 | I, O, U   | 321 EE  |
| 1  |                  | 2021-02-04 19:20 | Install   | 496 EE  |

## Next steps

For additional cloud-init examples of configuration changes, see the following:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Use cloud-init to add a user to a Linux VM in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use [cloud-init](#) to add a user on a virtual machine (VM) or virtual machine scale sets (VMSS) at provisioning time in Azure. This cloud-init script runs on first boot once the resources have been provisioned by Azure. For more information about how cloud-init works natively in Azure and the supported Linux distros, see [cloud-init overview](#).

## Add a user to a VM with cloud-init

One of the first tasks on any new Linux VM is to add an additional user for yourself to avoid the use of `root`. SSH keys are best practice for security and usability. Keys are added to the `~/.ssh/authorized_keys` file with this cloud-init script.

To add a user to a Linux VM, create a file in your current shell named `cloud_init_add_user.txt` and paste the following configuration. For this example, create the file in the Cloud Shell not on your local machine. You can use any editor you wish. Enter `sensible-editor cloud_init_add_user.txt` to create the file and see a list of available editors. Choose #1 to use the `nano` editor. Make sure that the whole cloud-init file is copied correctly, especially the first line. You need to provide your own public key (such as the contents of `~/.ssh/id_rsa.pub`) for the value of `ssh-authorized-keys:` - it has been shortened here to simplify the example.

```
#cloud-config
users:
  - default
  - name: myadminuser
    groups: sudo
    shell: /bin/bash
    sudo: ['ALL=(ALL) NOPASSWD:ALL']
    ssh-authorized-keys:
      - ssh-rsa AAAAB3<snip>
```

### NOTE

The `#cloud-config` file includes the `- default` parameter included. This will append the user, to the existing admin user created during provisioning. If you create a user without the `- default` parameter - the auto generated admin user created by the Azure platform would be overwritten.

Before deploying this image, you need to create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named `myResourceGroup` in the `eastus` location.

```
az group create --name myResourceGroup --location eastus
```

Now, create a VM with [az vm create](#) and specify the cloud-init file with `--custom-data cloud_init_add_user.txt` as follows:

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS:7-CI:latest \
--custom-data cloud_init_add_user.txt \
--generate-ssh-keys
```

SSH to the public IP address of your VM shown in the output from the preceding command. Enter your own `publicIpAddress` as follows:

```
ssh <publicIpAddress>
```

To confirm your user was added to the VM and the specified groups, view the contents of the `/etc/group` file as follows:

```
cat /etc/group
```

The following example output shows the user from the `cloud_init_add_user.txt` file has been added to the VM and the appropriate group:

```
root:x:0:
<snip />
sudo:x:27:myadminuser
<snip />
myadminuser:x:1000:
```

## Next steps

For additional cloud-init examples of configuration changes, see the following:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Use cloud-init to configure a swap partition on a Linux VM

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use [cloud-init](#) to configure the swap partition on various Linux distributions. The swap partition was traditionally configured by the Linux Agent (WALA) based on which distributions required one. This document will outline the process for building the swap partition on demand during provisioning time using cloud-init. For more information about how cloud-init works natively in Azure and the supported Linux distros, see [cloud-init overview](#)

## Create swap partition for Ubuntu based images

By default on Azure, Ubuntu gallery images do not create swap partitions. To enable swap partition configuration during VM provisioning time using cloud-init - please see the [AzureSwapPartitions document](#) on the Ubuntu wiki.

## Create swap partition for Red Hat and CentOS based images

Create a file in your current shell named `cloud_init_swappart.txt` and paste the following configuration. For this example, create the file in the Cloud Shell not on your local machine. You can use any editor you wish. Enter `sensible-editor cloud_init_swappart.txt` to create the file and see a list of available editors. Choose #1 to use the `nano` editor. Make sure that the whole cloud-init file is copied correctly, especially the first line.

```
#cloud-config
disk_setup:
  ephemeral0:
    table_type: gpt
    layout: [66, [33,82]]
    overwrite: true
fs_setup:
  - device: ephemeral0.1
    filesystem: ext4
  - device: ephemeral0.2
    filesystem: swap
mounts:
  - ["ephemeral0.1", "/mnt"]
  - ["ephemeral0.2", "none", "swap", "sw,nofail,x-systemd.requires=cloud-init.service", "0", "0"]
```

The mount is created with the `nofail` option to ensure that the boot will continue even if the mount is not completed successfully.

Before deploying this image, you need to create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named `myResourceGroup` in the `eastus` location.

```
az group create --name myResourceGroup --location eastus
```

Now, create a VM with `az vm create` and specify the cloud-init file with `--custom-data cloud_init_swappart.txt` as follows:

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS:7-CI:latest \
--custom-data cloud_init_swappart.txt \
--generate-ssh-keys
```

## Verify swap partition was created

SSH to the public IP address of your VM shown in the output from the preceding command. Enter your own **publicIpAddress** as follows:

```
ssh <publicIpAddress>
```

Once you have SSH'ed into the vm, check if the swap partition was created

```
swapon -s
```

The output from this command should look like this:

| Filename  | Type      | Size    | Used | Priority |
|-----------|-----------|---------|------|----------|
| /dev/sdb2 | partition | 2494440 | 0    | -1       |

### NOTE

If you have an existing Azure image that has a swap partition configured and you want to change the swap partition configuration for new images, you should remove the existing swap partition. Please see 'Customize Images to provision by cloud-init' document for more details.

## Next steps

For additional cloud-init examples of configuration changes, see the following:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Use cloud-init to run a bash script in a Linux VM in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use [cloud-init](#) to run an existing bash script on a Linux virtual machine (VM) or virtual machine scale sets (VMSS) at provisioning time in Azure. These cloud-init scripts run on first boot once the resources have been provisioned by Azure. For more information about how cloud-init works natively in Azure and the supported Linux distros, see [cloud-init overview](#)

## Run a bash script with cloud-init

With cloud-init you do not need to convert your existing scripts into a cloud-config, cloud-init accepts multiple input types, one of which is a bash script.

If you have been using the Linux Custom Script Azure Extension to run your scripts, you can migrate them to use cloud-init. However, Azure Extensions have integrated reporting to alert to script failures, a cloud-init image deployment will NOT fail if the script fails.

To see this functionality in action, create a simple bash script for testing. Like the cloud-init `#cloud-config` file, this script must be local to where you will be running the AzureCLI commands to provision your virtual machine. For this example, create the file in the Cloud Shell not on your local machine. You can use any editor you wish. Enter `sensible-editor simple_bash.sh` to create the file and see a list of available editors. Choose #1 to use the `nano` editor. Make sure that the whole cloud-init file is copied correctly, especially the first line.

```
#!/bin/sh
echo "this has been written via cloud-init" + $(date) >> /tmp/myScript.txt
```

Before deploying this image, you need to create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named *myResourceGroup* in the *eastus* location.

```
az group create --name myResourceGroup --location eastus
```

Now, create a VM with [az vm create](#) and specify the bash script file with `--custom-data simple_bash.sh` as follows:

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS:7-CI:latest \
--custom-data simple_bash.sh \
--generate-ssh-keys
```

## Verify bash script has run

SSH to the public IP address of your VM shown in the output from the preceding command. Enter your own `publicIpAddress` as follows:

```
ssh <publicIpAddress>
```

Change to the **/tmp** directory and verify that myScript.txt file exists and has the appropriate text inside of it. If it does not, you can check the **/var/log/cloud-init.log** for more details. Search for the following entry:

```
Running config-scripts-user using lock Running command ['/var/lib/cloud/instance/scripts/part-001']
```

## Next steps

For additional cloud-init examples of configuration changes, see the following:

- [Add an additional Linux user to a VM](#)
- [Run a package manager to update existing packages on first boot](#)
- [Change VM local hostname](#)
- [Install an application package, update configuration files and inject keys](#)

# Creating generalized images without a provisioning agent

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Microsoft Azure provides provisioning agents for Linux VMs in the form of the [walinuxagent](#) or [cloud-init](#) (recommended). But there could be a scenario when you don't want to use either of these applications for your provisioning agent, such as:

- Your Linux distro/version does not support cloud-init/Linux Agent.
- You require specific VM properties to be set, such as hostname.

## NOTE

If you do not require any properties to be set or any form of provisioning to happen you should consider creating a specialized image.

This article shows how you can setup your VM image to satisfy the Azure platform requirements and set the hostname, without installing a provisioning agent.

## Networking and reporting ready

In order to have your Linux VM communicating with Azure components, you will require a DHCP client to retrieve a host IP from the virtual network, as well as DNS resolution and route management. Most distros ship with these utilities out-of-the-box. Tools that have been tested on Azure by Linux distro vendors include

`dhclient` , `network-manager` , `systemd-networkd` and others.

## NOTE

Currently creating generalized images without a provisioning agent only supports DHCP-enabled VMs.

After networking has been setup and configured, you must "report ready". This will tell Azure that the VM has been successfully provisioning.

## IMPORTANT

Failing to report ready to Azure will result in your VM being rebooted!

## Demo/sample

This demo will show how you can take an existing Marketplace image (in this case, a Debian Buster VM) and remove the Linux Agent (walinuxagent), but also creating the most basic process to report to Azure that the VM is "ready".

### Create the resource group and base VM:

```
$ az group create --location eastus --name demo1
```

Create the base VM:

```
$ az vm create \
--resource-group demo1 \
--name demo1 \
--location eastus \
--ssh-key-value <ssh_pub_key_path> \
--public-ip-address-dns-name demo1 \
--image "debian:debian-10:10:latest"
```

## Remove the image provisioning Agent

Once the VM is provisioning, you can SSH into it and remove the Linux Agent:

```
$ sudo apt purge -y waagent
$ sudo rm -rf /var/lib/waagent /etc/waagent.conf /var/log/waagent.log
```

## Add required code to the VM

Also inside the VM, because we've removed the Azure Linux Agent we need to provide a mechanism to report ready.

### Python script

```
import http.client
import sys
from xml.etree import ElementTree

wireserver_ip = '168.63.129.16'
wireserver_conn = http.client.HTTPConnection(wireserver_ip)

print('Retrieving goal state from the Wireserver')
wireserver_conn.request(
    'GET',
    '/machine?comp=goalstate',
    headers={'x-ms-version': '2012-11-30'}
)

resp = wireserver_conn.getresponse()

if resp.status != 200:
    print('Unable to connect with wireserver')
    sys.exit(1)

wireserver_goalstate = resp.read().decode('utf-8')

xml_el = ElementTree.fromstring(wireserver_goalstate)

container_id = xml_el.findtext('Container/ContainerId')
instance_id = xml_el.findtext('Container/RoleInstanceList/RoleInstance/InstanceId')
incarnation = xml_el.findtext('Incarnation')
print(f'ContainerId: {container_id}')
print(f'InstanceId: {instance_id}')
print(f'Incarnation: {incarnation}')

# Construct the XML response we need to send to Wireserver to report ready.
health = ElementTree.Element('Health')
goalstate_incarnation = ElementTree.SubElement(health, 'GoalStateIncarnation')
goalstate_incarnation.text = incarnation
container = ElementTree.SubElement(health, 'Container')
container_id_el = ElementTree.SubElement(container, 'ContainerId')
container_id_el.text = container_id
role_instance_list = ElementTree.SubElement(container, 'RoleInstanceList')
role = ElementTree.SubElement(role_instance_list, 'Role')
instance_id_el = ElementTree.SubElement(role, 'InstanceId')
```

```
instance_id_el.text = instance_id
health_second = ElementTree.SubElement(role, 'Health')
state = ElementTree.SubElement(health_second, 'State')
state.text = 'Ready'

out_xml = ElementTree.tostring(
    health,
    encoding='unicode',
    method='xml'
)
print('Sending the following data to Wireserver:')
print(out_xml)

wireserver_conn.request(
    'POST',
    '/machine?comp=health',
    headers={
        'x-ms-version': '2012-11-30',
        'Content-Type': 'text/xml; charset=utf-8',
        'x-ms-agent-name': 'custom-provisioning'
    },
    body=out_xml
)

resp = wireserver_conn.getresponse()
print(f'Response: {resp.status} {resp.reason}')

wireserver_conn.close()
```

### Bash script

```

#!/bin/bash

attempts=1
until [ "$attempts" -gt 5 ]
do
    echo "obtaining goal state - attempt $attempts"
    goalstate=$(curl --fail -v -X 'GET' -H "x-ms-agent-name: azure-vm-register" \
        -H "Content-Type: text/xml;charset=utf-8" \
        -H "x-ms-version: 2012-11-30" \
        "http://168.63.129.16/machine/?comp=goalstate")
    if [ $? -eq 0 ]
    then
        echo "successfully retrieved goal state"
        retrieved_goal_state=true
        break
    fi
    sleep 5
    attempts=$((attempts+1))
done

if [ "$retrieved_goal_state" != "true" ]
then
    echo "failed to obtain goal state - cannot register this VM"
    exit 1
fi

container_id=$(grep ContainerId <<< "$goalstate" | sed 's/\s*<\/\*ContainerId>//g' | sed 's/\r//')
instance_id=$(grep InstanceId <<< "$goalstate" | sed 's/\s*<\/\*InstanceId>//g' | sed 's/\r//')

ready_doc=$(cat << EOF
<?xml version="1.0" encoding="utf-8"?>
<Health xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GoalStateIncarnation>1</GoalStateIncarnation>
    <Container>
        <ContainerId>$container_id</ContainerId>
        <RoleInstanceList>
            <Role>
                <InstanceId>$instance_id</InstanceId>
                <Health>
                    <State>Ready</State>
                </Health>
            </Role>
        </RoleInstanceList>
    </Container>
</Health>
EOF
)

attempts=1
until [ "$attempts" -gt 5 ]
do
    echo "registering with Azure - attempt $attempts"
    curl --fail -v -X 'POST' -H "x-ms-agent-name: azure-vm-register" \
        -H "Content-Type: text/xml;charset=utf-8" \
        -H "x-ms-version: 2012-11-30" \
        -d "$ready_doc" \
        "http://168.63.129.16/machine?comp=health"
    if [ $? -eq 0 ]
    then
        echo "successfully register with Azure"
        break
    fi
    sleep 5 # sleep to prevent throttling from wire server
done

```

#### Generic steps (if not using Python or Bash)

If your VM doesn't have Python installed or available, you can programmatically reproduce this above script logic with the following steps:

1. Retrieve the `ContainerId`, `InstanceId`, and `Incarnation` by parsing the response from the WireServer:

```
curl -X GET -H 'x-ms-version: 2012-11-30' http://168.63.129.16/machine?comp=goalstate .
```

2. Construct the following XML data, injecting the parsed `ContainerId`, `InstanceId`, and `Incarnation` from the above step:

```
<Health>
  <GoalStateIncarnation>INCARNATION</GoalStateIncarnation>
  <Container>
    <ContainerId>CONTAINER_ID</ContainerId>
    <RoleInstanceList>
      <Role>
        <InstanceId>INSTANCE_ID</InstanceId>
        <Health>
          <State>Ready</State>
        </Health>
      </Role>
    </RoleInstanceList>
  </Container>
</Health>
```

3. Post this data to WireServer:

```
curl -X POST -H 'x-ms-version: 2012-11-30' -H "x-ms-agent-name: WALinuxAgent" -H "Content-Type: text/xml; charset=utf-8" -d "$REPORT_READY_XML" http://168.63.129.16/machine?comp=health
```

## Automating running the code at first boot

This demo uses systemd, which is the most common init system in modern Linux distros. So the easiest and most native way to ensure this report ready mechanism runs at the right time is to create a systemd service unit. You can add the following unit file to `/etc/systemd/system` (this example names the unit file

```
azure-provisioning.service ):
```

```
[Unit]
Description=Azure Provisioning

[Service]
Type=oneshot
ExecStart=/usr/bin/python3 /usr/local/azure-provisioning.py
ExecStart=/bin/bash -c "hostnamectl set-hostname $(curl \
-H 'metadata: true' \
'http://169.254.169.254/metadata/instance/compute/name?api-version=2019-06-01&format=text')"
ExecStart=/usr/bin/systemctl disable azure-provisioning.service

[Install]
WantedBy=multi-user.target
```

This systemd service does three things for basic provisioning:

1. Reports ready to Azure (to indicate that it came up successfully).
2. Renames the VM based off of the user-supplied VM name by pulling this data from [Azure Instance Metadata Service \(IMDS\)](#). Note IMDS also provides other [instance metadata](#), such as SSH Public Keys, so you can set more than the hostname.
3. Disables itself so that it only runs on first boot and not on subsequent reboots.

With the unit on the filesystem, run the following to enable it:

```
$ sudo systemctl enable azure-provisioning.service
```

Now the VM is ready to be generalized and have an image created from it.

#### Completing the preparation of the image

Back on your development machine, run the following to prepare for image creation from the base VM:

```
$ az vm deallocate --resource-group demo1 --name demo1
$ az vm generalize --resource-group demo1 --name demo1
```

And create the image from this VM:

```
$ az image create \
--resource-group demo1 \
--source demo1 \
--location eastus \
--name demo1img
```

Now we are ready to create a new VM (or multiple VMs) from the image:

```
$ IMAGE_ID=$(az image show -g demo1 -n demo1img --query id -o tsv)
$ az vm create \
--resource-group demo12 \
--name demo12 \
--location eastus \
--ssh-key-value <ssh_pub_key_path> \
--public-ip-address-dns-name demo12 \
--image "$IMAGE_ID"
--enable-agent false
```

#### NOTE

It is important to set `--enable-agent` to `false` because walinuxagent doesn't exist on this VM that is going to be created from the image.

This VM should provisioning successfully. Logging into the newly-provisioning VM, you should be able to see the output of the report ready systemd service:

```
$ sudo journalctl -u azure-provisioning.service
-- Logs begin at Thu 2020-06-11 20:28:45 UTC, end at Thu 2020-06-11 20:31:24 UTC. --
Jun 11 20:28:49 thstringnopa systemd[1]: Starting Azure Provisioning...
Jun 11 20:28:54 thstringnopa python3[320]: Retrieving goal state from the Wireserver
Jun 11 20:28:54 thstringnopa python3[320]: ContainerId: 7b324f53-983a-43bc-b919-1775d6077608
Jun 11 20:28:54 thstringnopa python3[320]: InstanceId: fbb84507-46cd-4f4e-bd78-a2edaa9d059b._thstringnopa2
Jun 11 20:28:54 thstringnopa python3[320]: Sending the following data to Wireserver:
Jun 11 20:28:54 thstringnopa python3[320]: <Health><GoalStateIncarnation>1</GoalStateIncarnation><Container>
<ContainerId>7b324f53-983a-43bc-b919-1775d6077608</ContainerId><RoleInstanceList><Role><InstanceId>fbb84507-
46cd-4f4e-bd78-a2edaa9d059b._thstringnopa2</InstanceId><Health><State>Ready</State></Health></Role>
</RoleInstanceList></Container></Health>
Jun 11 20:28:54 thstringnopa python3[320]: Response: 200 OK
Jun 11 20:28:56 thstringnopa bash[472]: % Total    % Received % Xferd  Average Speed   Time     Time
Time   Current
Jun 11 20:28:56 thstringnopa bash[472]:   Dload  Upload   Total   Spent
Left  Speed
Jun 11 20:28:56 thstringnopa bash[472]: [158B blob data]
Jun 11 20:28:56 thstringnopa2 systemctl[475]: Removed /etc/systemd/system/multi-user.target.wants/azure-
provisioning.service.
Jun 11 20:28:56 thstringnopa2 systemd[1]: azure-provisioning.service: Succeeded.
Jun 11 20:28:56 thstringnopa2 systemd[1]: Started Azure Provisioning.
```

## Support

If you implement your own provisioning code/agent, then you own the support of this code, Microsoft support will only investigate issues relating to the provisioning interfaces not being available. We are continually making improvements and changes in this area, so you must monitor for changes in cloud-init and Azure Linux Agent for provisioning API changes.

## Next steps

For more information, see [Linux provisioning](#).

# Disable or remove the Linux Agent from VMs and images

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Before removing the Linux Agent, you must understand of what VM will not be able to do after the Linux Agent is removed.

Azure virtual machine (VM) [extensions](#) are small applications that provide post-deployment configuration and automation tasks on Azure VMs, extensions are installed and managed by the Azure control plane. It is the job of the [Azure Linux Agent](#) to process the platform extension commands and ensure the correct state of the extension inside the VM.

The Azure platform hosts many extensions that range from VM configuration, monitoring, security, and utility applications. There is a large choice of first and third-party extensions, examples of key scenarios that extensions are used for:

- Supporting first party Azure services, such as Azure Backup, Monitoring, Disk Encryption, Security, Site Replication and others.
- SSH / Password resets
- VM configuration - Running custom scripts, installing Chef, Puppet agents etc..
- Third-party products, such as AV products, VM vulnerability tools, VM and App monitoring tooling.
- Extensions can be bundled with a new VM deployment. For example, they can be part of a larger deployment, configuring applications on VM provision, or run against any supported extension operated systems post deployment.

## Disabling extension processing

There are several ways to disable extension processing, depending on your needs, but before you continue, you **MUST** remove all extensions deployed to the VM, for example using the Azure CLI, you can [list](#) and [delete](#):

```
az vm extension delete -g MyResourceGroup --vm-name MyVm -n extension_name
```

### NOTE

If you do not do the above, the platform will try to send the extension configuration and timeout after 40min.

### Disable at the control plane

If you are not sure whether you will need extensions in the future, you can leave the Linux Agent installed on the VM, then disable extension processing capability from the platform. This is option is available in

`Microsoft.Compute` api version `2018-06-01` or higher, and does not have a dependency on the Linux Agent version installed.

```
az vm update -g <resourceGroup> -n <vmName> --set osProfile.allowExtensionOperations=false
```

You can easily reenable this extension processing from the platform, with the above command, but set it to 'true'.

# Remove the Linux Agent from a running VM

Ensure you have **removed** all existing extensions from the VM before, as per above.

## Step 1: Remove the Azure Linux Agent

If you just remove the Linux Agent, and not the associated configuration artifacts, you can reinstall at a later date. Run one of the following, as root, to remove the Azure Linux Agent:

### For Ubuntu >=18.04

```
apt -y remove walinuxagent
```

### For Redhat >= 7.7

```
yum -y remove WALinuxAgent
```

### For SUSE

```
zypper --non-interactive remove python-azure-agent
```

## Step 2: (Optional) Remove the Azure Linux Agent artifacts

### IMPORTANT

You can remove all associated artifacts of the Linux Agent, but this will mean you cannot reinstall it at a later date. Therefore, it is strongly recommended you consider disabling the Linux Agent first, removing the Linux Agent using the above only.

If you know you will not ever reinstall the Linux Agent again, then you can run the following:

### For Ubuntu >=18.04

```
apt -y purge walinuxagent
rm -rf /var/lib/waagent
rm -f /var/log/waagent.log
```

### For Redhat >= 7.7

```
yum -y remove WALinuxAgent
rm -f /etc/waagent.conf.rpmsave
rm -rf /var/lib/waagent
rm -f /var/log/waagent.log
```

### For SUSE

```
zypper --non-interactive remove python-azure-agent
rm -f /etc/waagent.conf.rpmsave
rm -rf /var/lib/waagent
rm -f /var/log/waagent.log
```

# Preparing an image without the Linux Agent

If you have an image that already contains cloud-init, and you want to remove the Linux agent, but still provision using cloud-init, run the steps in Step 2 (and optionally Step 3) as root to remove the Azure Linux Agent and

then the following will remove the cloud-init configuration and cached data, and prepare the VM to create a custom image.

```
cloud-init clean --logs --seed
```

## Deprovision and create an image

The Linux Agent has the ability to clean up some of the existing image metadata, with the step "waagent - deprovision+user", however, after it has been removed, you will need to perform actions such as the below, and remove any other sensitive data from it.

- Remove all existing ssh host keys

```
rm /etc/ssh/ssh_host_*key*
```

- Delete the admin account

```
touch /var/run/utmp
userdel -f -r <admin_user_account>
```

- Delete the root password

```
passwd -d root
```

Once you have completed the above, you can create the custom image using the Azure CLI.

### Create a regular managed image

```
az vm deallocate -g <resource_group> -n <vm_name>
az vm generalize -g <resource_group> -n <vm_name>
az image create -g <resource_group> -n <image_name> --source <vm_name>
```

### Create an image version in a Azure Compute Gallery

```
az sig image-version create \
-g $sigResourceGroup
--gallery-name $signName
--gallery-image-definition $imageDefName
--gallery-image-version 1.0.0
--managed-image /subscriptions/00000000-0000-0000-0000-
00000000xxxx/resourceGroups/imageGroups/providers/images/MyManagedImage
```

### Creating a VM from an image that does not contain a Linux Agent

When you create the VM from the image with no Linux Agent, you need to ensure the VM deployment configuration indicates extensions are not supported on this VM.

#### NOTE

If you do not do the above, the platform will try to send the extension configuration and timeout after 40min.

To deploy the VM with extensions disabled, you can use the Azure CLI with [--enable-agent](#).

```
az vm create \
--resource-group $resourceGroup \
--name $prodVmName \
--image RedHat:RHEL:8.1-ci:latest \
--admin-username azadmin \
--ssh-key-value "$sshPubkeyPath" \
--enable-agent false
```

Alternatively, you can do this using Azure Resource Manager (ARM) templates, by setting

```
"provisionVMAgent": false,
```

```
"osProfile": {
  "computerName": "[parameters('virtualMachineName')]",
  "adminUsername": "[parameters('adminUsername')]",
  "linuxConfiguration": {
    "disablePasswordAuthentication": "true",
    "provisionVMAgent": false,
    "ssh": {
      "publicKeys": [
        {
          "path": "[concat('/home/', parameters('adminUsername'), '/.ssh/authorized_keys')]",
          "keyData": "[parameters('adminPublicKey')]"
        }
      ]
    }
  }
}
```

## Next steps

For more information, see [Provisioning Linux](#).

# Prepare a Windows VHD or VHDX to upload to Azure

9/21/2022 • 20 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Before you upload a Windows virtual machine (VM) from on-premises to Azure, you must prepare the virtual hard disk (VHD or VHDX). Azure supports both generation 1 and generation 2 VMs that are in VHD file format and that have a fixed-size disk. The maximum size allowed for the OS VHD on a generation 1 VM is 2 TB.

You can convert a VHDX file to VHD, convert a dynamically expanding disk to a fixed-size disk, but you can't change a VM's generation. For more information, see [Should I create a generation 1 or 2 VM in Hyper-V?](#) and [Support for generation 2 VMs on Azure](#).

For information about the support policy for Azure VMs, see [Microsoft server software support for Azure VMs](#).

## NOTE

The instructions in this article apply to:

- The 64-bit version of Windows Server 2008 R2 and later Windows Server operating systems. For information about running a 32-bit operating system in Azure, see [Support for 32-bit operating systems in Azure VMs](#).
- If any Disaster Recovery tool will be used to migrate the workload, like Azure Site Recovery or Azure Migrate, this process is still required on the Guest OS to prepare the image before the migration.

## System File Checker

### Run Windows System File Checker utility before generalization of OS image

The System File Checker (SFC) is used to verify and replace Windows system files.

## IMPORTANT

Use an elevated PowerShell session to run the examples in this article.

Run the SFC command:

```
sfc.exe /scannow
```

```
Beginning system scan. This process will take some time.
```

```
Beginning verification phase of system scan.  
Verification 100% complete.
```

```
Windows Resource Protection did not find any integrity violations.
```

After the SFC scan completes, install Windows Updates and restart the computer.

## Set Windows configurations for Azure

#### NOTE

Azure platform mounts an ISO file to the DVD-ROM when a Windows VM is created from a generalized image. For this reason, the DVD-ROM must be enabled in the OS in the generalized image. If it is disabled, the Windows VM will be stuck at out-of-box experience (OOBE).

#### 1. Remove any static persistent routes in the routing table:

- To view the routing table, run `route.exe print`.
- Check the **Persistence Routes** section. If there's a persistent route, use the `route.exe delete` command to remove it.

#### 2. Remove the WinHTTP proxy:

```
netsh.exe winhttp reset proxy
```

If the VM needs to work with a specific proxy, add a proxy exception for the Azure IP address ([168.63.129.16](#)) so the VM can connect to Azure:

```
$proxyAddress='<your proxy server>'  
$proxyBypassList='<your list of bypasses>;168.63.129.16'  
netsh.exe winhttp set proxy $proxyAddress $proxyBypassList
```

#### 3. Open DiskPart:

```
diskpart.exe
```

Set the disk SAN policy to [Onlineall](#):

```
DISKPART> san policy=onlineall  
DISKPART> exit
```

#### 4. Set Coordinated Universal Time (UTC) time for Windows. Also, set the startup type of the Windows time service **w32time** to **Automatic**:

```
Set-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Control\TimeZoneInformation -Name  
RealTimeIsUniversal -Value 1 -Type DWord -Force  
Set-Service -Name w32time -StartupType Automatic
```

#### 5. Set the power profile to high performance:

```
powercfg.exe /setactive SCHEME_MIN  
powercfg /setacvalueindex SCHEME_CURRENT SUB_VIDEO VIDEOIDLE 0
```

#### 6. Make sure the environmental variables **TEMP** and **TMP** are set to their default values:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Environment' -Name  
TEMP -Value "%SystemRoot%\TEMP" -Type ExpandString -Force  
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Environment' -Name TMP  
-Value "%SystemRoot%\TEMP" -Type ExpandString -Force
```

#### 7. For VMs with legacy operating systems (Windows Server 2012 R2 or Windows 8.1 and below), make

sure the latest Hyper-V Integration Component Services are installed. For more information, see [Hyper-V integration components update for Windows VM](#).

#### NOTE

In a scenario where VMs are to be set up with a disaster recovery solution between the on-premises VMware server and Azure, the Hyper-V Integration Component Services can't be used. If that's the case, please contact the VMware support to migrate the VM to Azure and make it co-reside in VMware server.

## Check the Windows services

Make sure that each of the following Windows services is set to the Windows default value. These services are the minimum that must be configured to ensure VM connectivity. To set the startup settings, run the following example:

```
Get-Service -Name BFE, Dhcp, Dnscache, IKEEXT, iphlpsvc, nsi, mpssvc, RemoteRegistry |  
    Where-Object StartType -ne Automatic |  
        Set-Service -StartupType Automatic  
  
Get-Service -Name Netlogon, Netman, TermService |  
    Where-Object StartType -ne Manual |  
        Set-Service -StartupType Manual
```

## Update remote desktop registry settings

Make sure the following settings are configured correctly for remote access:

#### NOTE

If you receive an error message when running

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\ Policies\Microsoft\Windows NT\Terminal Services' -Name <string> -  
Value <object>
```

, you can safely ignore it. It means the domain isn't setting that configuration through a Group Policy Object.

1. Remote Desktop Protocol (RDP) is enabled:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server' -Name  
fDenyTSConnections -Value 0 -Type DWord -Force  
Set-ItemProperty -Path 'HKLM:\SOFTWARE\ Policies\Microsoft\Windows NT\Terminal Services' -Name  
fDenyTSConnections -Value 0 -Type DWord -Force
```

2. The RDP port is set up correctly using the default port of 3389:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -  
Name PortNumber -Value 3389 -Type DWord -Force
```

When you deploy a VM, the default rules are created for port 3389. To change the port number, do that after the VM is deployed in Azure.

3. The listener is listening on every network interface:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -  
Name LanAdapter -Value 0 -Type DWord -Force
```

4. Configure network-level authentication (NLA) mode for the RDP connections:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp' -Name UserAuthentication -Value 1 -Type DWord -Force
```

5. Set the keep-alive value:

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services' -Name KeepAliveEnable -Value 1 -Type DWord -Force  
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services' -Name KeepAliveInterval -Value 1 -Type DWord -Force  
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -Name KeepAliveTimeout -Value 1 -Type DWord -Force
```

6. Set the reconnect options:

```
Set-ItemProperty -Path 'HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services' -Name fDisableAutoReconnect -Value 0 -Type DWord -Force  
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -Name fInheritReconnectSame -Value 1 -Type DWord -Force  
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -Name fReconnectSame -Value 0 -Type DWord -Force
```

7. Limit the number of concurrent connections:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp' -Name MaxInstanceCount -Value 4294967295 -Type DWord -Force
```

8. Remove any self-signed certificates tied to the RDP listener:

```
if ((Get-Item -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp').Property -contains 'SSLCertificateSHA1Hash')  
{  
    Remove-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-Tcp' -Name SSLCertificateSHA1Hash -Force  
}
```

This code ensures that you can connect when you deploy the VM. You can also review these settings after the VM is deployed in Azure.

9. If the VM is part of a domain, check the following policies to make sure the previous settings aren't reverted.

GOAL	POLICY	VALUE
RDP is enabled	Computer Configuration\Policies\Windows Settings\Administrative Templates\Components\Remote Desktop Services\Remote Desktop Session Host\Connections	Allow users to connect remotely by using Remote Desktop
NLA group policy	Settings\Administrative Templates\Components\Remote Desktop Services\Remote Desktop Session Host\Security	Require user authentication for remote access by using NLA

GOAL	POLICY	VALUE
Keep-alive settings	Computer Configuration\Policies\Windows Settings\Administrative Templates\Windows Components\Remote Desktop Services\Remote Desktop Session Host\Connections	Configure keep-alive connection interval
Reconnect settings	Computer Configuration\Policies\Windows Settings\Administrative Templates\Windows Components\Remote Desktop Services\Remote Desktop Session Host\Connections	Reconnect automatically
Limited number of connection settings	Computer Configuration\Policies\Windows Settings\Administrative Templates\Windows Components\Remote Desktop Services\Remote Desktop Session Host\Connections	Limit number of connections

## Configure Windows Firewall rules

- Turn on Windows Firewall on the three profiles (domain, standard, and public):

```
Set-NetFirewallProfile -Profile Domain, Public, Private -Enabled True
```

- Run the following example to allow WinRM through the three firewall profiles (domain, private, and public), and enable the PowerShell remote service:

```
Enable-PSRemoting -Force
Set-NetFirewallRule -Name WINRM-HTTP-In-TCP, WINRM-HTTP-In-TCP-PUBLIC -Enabled True
```

- Enable the following firewall rules to allow the RDP traffic:

```
Set-NetFirewallRule -Group '@FirewallAPI.dll,-28752' -Enabled True
```

- Enable the rule for file and printer sharing so the VM can respond to ping requests inside the virtual network:

```
Set-NetFirewallRule -Name FPS-ICMP4-ERQ-In -Enabled True
```

- Create a rule for the Azure platform network:

```
New-NetFirewallRule -DisplayName AzurePlatform -Direction Inbound -RemoteAddress 168.63.129.16 -Profile Any -Action Allow -EdgeTraversalPolicy Allow
New-NetFirewallRule -DisplayName AzurePlatform -Direction Outbound -RemoteAddress 168.63.129.16 -Profile Any -Action Allow
```

6. If the VM is part of a domain, check the following Azure AD policies to make sure the previous settings aren't reverted.

GOAL	POLICY	VALUE
Enable the Windows Firewall profiles	Computer Configuration\Policies\Windows Settings\Administrative Templates\Network\Network Connection\Windows Firewall\Domain Profile\Windows Firewall	Protect all network connections
Enable RDP	Computer Configuration\Policies\Windows Settings\Administrative Templates\Network\Network Connection\Windows Firewall\Domain Profile\Windows Firewall	Allow inbound Remote Desktop exceptions
	Computer Configuration\Policies\Windows Settings\Administrative Templates\Network\Network Connection\Windows Firewall\Standard Profile\Windows Firewall	Allow inbound Remote Desktop exceptions
Enable ICMP-V4	Computer Configuration\Policies\Windows Settings\Administrative Templates\Network\Network Connection\Windows Firewall\Domain Profile\Windows Firewall	Allow ICMP exceptions
	Computer Configuration\Policies\Windows Settings\Administrative Templates\Network\Network Connection\Windows Firewall\Standard Profile\Windows Firewall	Allow ICMP exceptions

## Verify the VM

Make sure the VM is healthy, secure, and RDP accessible:

1. To make sure the disk is healthy and consistent, check the disk at the next VM restart:

```
chkdsk.exe /f
```

Make sure the report shows a clean and healthy disk.

2. Set the Boot Configuration Data (BCD) settings.

```

cmd

bcdedit.exe /set "{bootmgr}" integrityservices enable
bcdedit.exe /set "{default}" device partition=C:
bcdedit.exe /set "{default}" integrityservices enable
bcdedit.exe /set "{default}" recoveryenabled Off
bcdedit.exe /set "{default}" osdevice partition=C:
bcdedit.exe /set "{default}" bootstatuspolicy IgnoreAllFailures

#Enable Serial Console Feature
bcdedit.exe /set "{bootmgr}" displaybootmenu yes
bcdedit.exe /set "{bootmgr}" timeout 5
bcdedit.exe /set "{bootmgr}" bootechos yes
bcdedit.exe /ems "{current}" ON
bcdedit.exe /emssettings EMSPORT:1 EMSBAUDRATE:115200

exit

```

3. The dump log can be helpful in troubleshooting Windows crash issues. Enable the dump log collection:

```

# Set up the guest OS to collect a kernel dump on an OS crash event
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\CrashControl' -Name CrashDumpEnabled -Type DWord -Force -Value 2
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\CrashControl' -Name DumpFile -Type ExpandString -Force -Value "%SystemRoot%\MEMORY.DMP"
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\CrashControl' -Name NMICrashDump -Type DWord -Force -Value 1

# Set up the guest OS to collect user mode dumps on a service crash event
$key = 'HKLM:\SOFTWARE\Microsoft\Windows\Windows Error Reporting\LocalDumps'
if ((Test-Path -Path $key) -eq $false) {(New-Item -Path 'HKLM:\SOFTWARE\Microsoft\Windows\Windows Error Reporting' -Name LocalDumps)}
New-ItemProperty -Path $key -Name DumpFolder -Type ExpandString -Force -Value 'C:\CrashDumps'
New-ItemProperty -Path $key -Name CrashCount -Type DWord -Force -Value 10
New-ItemProperty -Path $key -Name DumpType -Type DWord -Force -Value 2
Set-Service -Name WerSvc -StartupType Manual

```

4. Verify that the Windows Management Instrumentation (WMI) repository is consistent:

```
winmgmt.exe /verifyrepository
```

If the repository is corrupted, see [WMI: Repository corruption or not.](#)

5. Make sure no other applications than TermService are using port 3389. This port is used for the RDP service in Azure. To see which ports are used on the VM, run `netstat.exe -anob`:

```
netstat.exe -anob
```

The following is an example.

```

netstat.exe -anob | findstr 3389
TCP    0.0.0.0:3389          0.0.0.0:0          LISTENING      4056
TCP    [::]:3389             [::]:0            LISTENING      4056
UDP    0.0.0.0:3389          *:*              4056
UDP    [::]:3389             *:*              4056

tasklist /svc | findstr 4056
svchost.exe           4056  TermService

```

6. To upload a Windows VHD that's a domain controller:

- Follow [these extra steps](#) to prepare the disk.
- Make sure you know the Directory Services Restore Mode (DSRM) password in case you ever have to start the VM in DSRM. For more information, see [Set a DSRM password](#).

7. Make sure you know the built-in administrator account and password. You might want to reset the current local administrator password and make sure you can use this account to sign in to Windows through the RDP connection. This access permission is controlled by the "Allow log on through Remote Desktop Services" Group Policy Object. View this object in the Local Group Policy Editor:

- Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment

8. Check the following Azure AD policies to make sure they're not blocking RDP access:

- Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment\Deny access to this computer from the network

- Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment\Deny log on through Remote Desktop Services

9. Check the following Azure AD policy to make sure they're not removing any of the required access accounts:

- Computer Configuration\Windows Settings\Security Settings\Local Policies\User Rights Assignment\Access this computer from the network

The policy should list the following groups:

- Administrators
- Backup Operators
- Everyone
- Users

10. Restart the VM to make sure that Windows is still healthy and can be reached through the RDP connection. At this point, consider creating a VM on your local Hyper-V server to make sure the VM starts completely. Then test to make sure you can reach the VM through RDP.

11. Remove any extra Transport Driver Interface (TDI) filters. For example, remove software that analyzes TCP packets or extra firewalls.

12. Uninstall any other third-party software or driver that's related to physical components or any other virtualization technology.

## Install Windows updates

### NOTE

To avoid an accidental reboot during the VM provisioning, we recommend completing all Windows update installations and to make sure there's no pending restart. One way to do this is to install all Windows updates and to reboot the VM before performing the migration to Azure.

If you also need to do a generalization of the OS (sysprep), you must update Windows and restart the VM before running the Sysprep command.

Ideally, you should keep the machine updated to the *patch level*, if this isn't possible, make sure the following updates are installed. To get the latest updates, see the Windows update history pages: [Windows 10](#), and [Windows Server 2019](#), [Windows 8.1](#), and [Windows Server 2012 R2](#) and [Windows 7 SP1](#), and [Windows Server](#)

## 2008 R2 SP1.

COMPONENT	BINARY	WINDOW S 7 SP1, WINDOW S SERVER 2008 R2 SP1	WINDOW S 8, WINDOW S SERVER 2012	WINDOW S 8.1, WINDOW S SERVER 2012 R2	WINDOW S 10 V1607, WINDOW S SERVER 2016 V1607	WINDOW S 10 V1703	WINDOW S 10, V1709, WINDOW S SERVER 2016 V1709	WINDOW S 10 V1803, WINDOW S SERVER 2016 V1803
Storage	disk.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 17638 / 6.2.9200. 21757 - KB31370 61	6.3.9600. 18203 - KB31370 61	-	-	-	-
	storport.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 17188 / 6.2.9200. 21306 - KB30184 89	6.3.9600. 18573 - KB40227 26	10.0.143 93.1358 - KB40227 15	10.0.150 63.332	-	-
	ntfs.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 17623 / 6.2.9200. 21743 - KB31212 55	6.3.9600. 18654 - KB40227 26	10.0.143 93.1198 - KB40227 15	10.0.150 63.447	-	-
	lologmsg.dll	6.1.7601. 23403 - KB31255 74	6.2.9200. 16384 - KB29953 87	-	-	-	-	-
	Classpnp.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 17061 / 6.2.9200. 21180 - KB29953 87	6.3.9600. 18334 - KB31726 14	10.0.143 93.953 - KB40227 15	-	-	-
	Volsnap.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 17047 / 6.2.9200. 21165 - KB29753 31	6.3.9600. 18265 - KB31453 84	-	10.0.150 63.0	-	-
	partmgr.sys	6.1.7601. 23403 - KB31255 74	6.2.9200. 16681 - KB28771 14	6.3.9600. 17401 - KB30008 50	10.0.143 93.953 - KB40227 15	10.0.150 63.0	-	-
	volmgr.sys					10.0.150 63.0	-	-

COMPONENT	BINARY	WINDOW S 7 SP1, WINDOW S SERVER 2008 R2 SP1	WINDOW S 8, WINDOW S SERVER 2012	WINDOW S 8.1, WINDOW S SERVER 2012 R2	WINDOW S 10 V1607, WINDOW S SERVER 2016 V1607	WINDOW S 10 V1703	WINDOW S 10 V1709, WINDOW S SERVER 2016 V1709	WINDOW S 10 V1803, WINDOW S SERVER 2016 V1803
	Volmgrx.sys	6.1.7601.23403 - KB3125574	-	-	-	10.0.15063.0	-	-
	Mscsi.sys	6.1.7601.23403 - KB3125574	6.2.9200.21006 - KB2955163	6.3.9600.18624 - KB4022726	10.0.14393.1066 - KB4022715	10.0.15063.447	-	-
	Msdsm.sys	6.1.7601.23403 - KB3125574	6.2.9200.21474 - KB3046101	6.3.9600.18592 - KB4022726	-	-	-	-
	Mpio.sys	6.1.7601.23403 - KB3125574	6.2.9200.21190 - KB3046101	6.3.9600.18616 - KB4022726	10.0.14393.1198 - KB4022715	-	-	-
	vmstorfl.sys	6.3.9600.18907 - KB4072650	6.3.9600.18080 - KB3063109	6.3.9600.18907 - KB4072650	10.0.14393.2007 - KB4345418	10.0.15063.850 - KB4345419	10.0.16299.371 - KB4345420	-
	Fveapi.dll	6.1.7601.23311 - KB3125574	6.2.9200.20930 - KB2930244	6.3.9600.18294 - KB3172614	10.0.14393.576 - KB4022715	-	-	-
	Fveapibase.dll	6.1.7601.23403 - KB3125574	6.2.9200.20930 - KB2930244	6.3.9600.17415 - KB3172614	10.0.14393.206 - KB4022715	-	-	-
Network	netvsc.sys	-	-	-	10.0.14393.1198 - KB4022715	10.0.15063.250 - KB4020001	-	-
	mrxsmb10.sys	6.1.7601.23816 - KB4022722	6.2.9200.22108 - KB4022724	6.3.9600.18603 - KB4022726	10.0.14393.479 - KB4022715	10.0.15063.483	-	-
	mrxsmb20.sys	6.1.7601.23816 - KB4022722	6.2.9200.21548 - KB4022724	6.3.9600.18586 - KB4022726	10.0.14393.953 - KB4022715	10.0.15063.483	-	-

COMPONENT	BINARY	WINDOW S 7 SP1, WINDOW S SERVER 2008 R2 SP1	WINDOW S 8, WINDOW S SERVER 2012	WINDOW S 8.1, WINDOW S SERVER 2012 R2	WINDOW S 10 V1607, WINDOW S SERVER 2016 V1607	WINDOW S 10 V1703	WINDOW S 10 V1709, WINDOW S SERVER 2016 V1709	WINDOW S 10 V1803, WINDOW S SERVER 2016 V1803
	mrxsmb.sys	6.1.7601.23816 - KB4022722	6.2.9200.22074 - KB4022724	6.3.9600.18586 - KB4022726	10.0.14393.953 - KB4022715	10.0.15063.0	-	-
	tcpip.sys	6.1.7601.23761 - KB4022722	6.2.9200.22070 - KB4022724	6.3.9600.18478 - KB4022726	10.0.14393.1358 - KB4022715	10.0.15063.447	-	-
	http.sys	6.1.7601.23403 - KB3125574	6.2.9200.17285 - KB3042553	6.3.9600.18574 - KB4022726	10.0.14393.251 - KB4022715	10.0.15063.483	-	-
	vmswitch.sys	6.1.7601.23727 - KB4022719	6.2.9200.22117 - KB4022724	6.3.9600.18654 - KB4022726	10.0.14393.1358 - KB4022715	10.0.15063.138	-	-
Core	ntoskrnl.exe	6.1.7601.23807 - KB4022719	6.2.9200.22170 - KB4022718	6.3.9600.18696 - KB4022726	10.0.14393.1358 - KB4022715	10.0.15063.483	-	-
Remote Desktop Services	rdpcorets.dll	6.2.9200.21506 - KB4022719	6.2.9200.22104 - KB4022724	6.3.9600.18619 - KB4022726	10.0.14393.1198 - KB4022715	10.0.15063.0	-	-
	termsrv.dll	6.1.7601.23403 - KB3125574	6.2.9200.17048 - KB2973501	6.3.9600.17415 - KB3000850	10.0.14393.0 - KB4022715	10.0.15063.0	-	-
	termdd.sys	6.1.7601.23403 - KB3125574	-	-	-	-	-	-
	win32k.sys	6.1.7601.23807 - KB4022719	6.2.9200.22168 - KB4022718	6.3.9600.18698 - KB4022726	10.0.14393.594 - KB4022715	-	-	-
	rdpdd.dll	6.1.7601.23403 - KB3125574	-	-	-	-	-	-

COMPONENT	BINARY	WINDOW S 7 SP1, WINDOW S SERVER 2008 R2 SP1	WINDOW S 8, WINDOW S SERVER 2012	WINDOW S 8.1, WINDOW S SERVER 2012 R2	WINDOW S 10 V1607, WINDOW S SERVER 2016 V1607	WINDOW S 10 V1703	WINDOW S 10 V1709, WINDOW S SERVER 2016 V1709	WINDOW S 10 V1803, WINDOW S SERVER 2016 V1803
	rdpwd.sys	6.1.7601.23403 - KB3125574	-	-	-	-	-	-
Security	MS17-010	KB4012212	KB4012213	KB4012213	KB4012606	KB4012606	-	-
			KB4012216		KB4013198	KB4013198	-	-
		KB4012215	KB4012214	KB4012216	KB4013429	KB4013429	-	-
			KB4012217		KB4013429	KB4013429	-	-
	CVE-2018-0886	KB4103718	KB4103730	KB4103725	KB4103723	KB4103731	KB4103727	KB4103721
		KB4103712	KB4103726	KB4103715				

#### NOTE

To avoid an accidental reboot during VM provisioning, we recommend ensuring that all Windows Update installations are finished and that no updates are pending. One way to do this is to install all possible Windows updates and reboot once before you run the `sysprep.exe` command.

## Determine when to use Sysprep

System Preparation Tool (`sysprep.exe`) is a process you can run to reset a Windows installation. Sysprep provides an "out of the box" experience by removing all personal data and resetting several components.

You typically run `sysprep.exe` to create a template from which you can deploy several other VMs that have a specific configuration. The template is called a *generalized image*.

To create only one VM from one disk, you don't have to use Sysprep. Instead, you can create the VM from a *specialized image*. For information about how to create a VM from a specialized disk, see:

- [Create a VM from a specialized disk](#)
- [Create a VM from a specialized VHD disk](#)

To create a generalized image, you need to run Sysprep. For more information, see [How to use Sysprep: An introduction](#).

Not every role or application that's installed on a Windows-based computer supports generalized images. Before you use this procedure, make sure Sysprep supports the role of the computer. For more information, see

## Sysprep support for server roles.

In particular, Sysprep requires the drives to be fully decrypted before execution. If you have enabled encryption on your VM, disable it before running Sysprep.

### Generalize a VHD

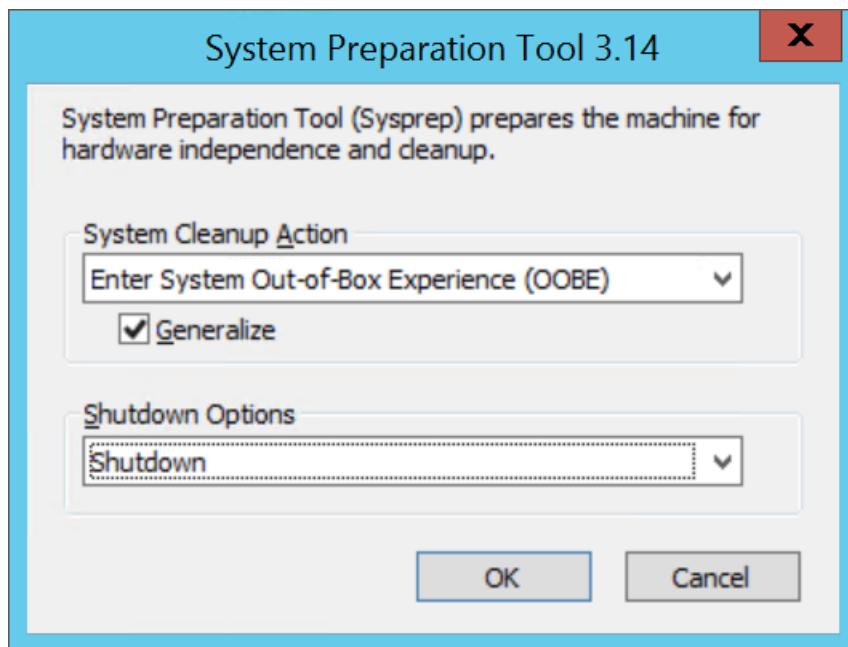
#### NOTE

If you're creating a generalized image from an existing Azure VM, we recommend to remove the VM extensions before running the sysprep.

#### NOTE

After you run `sysprep.exe` in the following steps, turn off the VM. Don't turn it back on until you create an image from it in Azure.

1. Sign in to the Windows VM.
2. Run a PowerShell session as an administrator.
3. Delete the panther directory (`C:\Windows\Panther`).
4. Change the directory to `%windir%\system32\sysprep`. Then run `sysprep.exe`.
5. In the **System Preparation Tool** dialog box, select **Enter System Out-of-Box Experience (OOBE)**, and make sure the **Generalize** checkbox is selected.



6. In **Shutdown Options**, select **Shutdown**.
7. Select **OK**.
8. When Sysprep finishes, shut down the VM. Don't use **Restart** to shut down the VM.

Now the VHD is ready to be uploaded. For more information about how to create a VM from a generalized disk, see [Upload a generalized VHD and use it to create a new VM in Azure](#).

#### **NOTE**

A custom *unattend.xml*/file is not supported. Although we do support the **additionalUnattendContent** property, that provides only limited support for adding [microsoft-windows-shell-setup](#) options into the *unattend.xml*/file that the Azure provisioning agent uses. You can use, for example, **additionalUnattendContent** to add FirstLogonCommands and LogonCommands. For more information, see [additionalUnattendContent FirstLogonCommands example](#).

## Convert the virtual disk to a fixed size VHD

#### **NOTE**

If you're going to use Azure PowerShell to [upload your disk to Azure](#) and you have [Hyper-V](#) enabled, this step is optional. [Add-AzVHD](#) will perform it for you.

Use one of the methods in this section to convert and resize your virtual disk to the required format for Azure:

1. Back up the VM before you run the virtual disk conversion or resize process.
2. Make sure that the Windows VHD works correctly on the local server. Resolve any errors within the VM itself before you try to convert or upload it to Azure.
3. Convert the virtual disk to type fixed.
4. Resize the virtual disk to meet Azure requirements:
  - a. Disks in Azure must have a virtual size aligned to 1 MiB. If your VHD is a fraction of 1 MiB, you'll need to resize the disk to a multiple of 1 MiB. Disks that are fractions of a MiB cause errors when creating images from the uploaded VHD. To verify the size you can use the PowerShell [Get-VHD](#) cmdlet to show "Size", which must be a multiple of 1 MiB in Azure, and "FileSize", which will be equal to "Size" plus 512 bytes for the VHD footer.

```
$vhd = Get-VHD -Path C:\test\MyNewVM.vhd  
$vhd.Size % 1MB  
0  
$vhd.FileSize - $vhd.Size  
512
```

- b. The maximum size allowed for the OS VHD with a generation 1 VM is 2,048 GiB (2 TiB).
- c. The maximum size for a data disk is 32,767 GiB (32 TiB).

#### **NOTE**

- If you are preparing a Windows OS disk after you convert to a fixed disk and resize if needed, create a VM that uses the disk. Start and sign in to the VM and continue with the sections in this article to finish preparing it for uploading.
- If you are preparing a data disk you may stop with this section and proceed to uploading your disk.

### Use Hyper-V Manager to convert the disk

1. Open Hyper-V Manager and select your local computer on the left. In the menu above the computer list, select **Action > Edit Disk**.
2. On the **Locate Virtual Hard Disk** page, select your virtual disk.
3. On the **Choose Action** page, select **Convert > Next**.
4. To convert from VHDX, select **VHD > Next**.

5. To convert from a dynamically expanding disk, select **Fixed size** > **Next**.
6. Locate and select a path to save the new VHD file.
7. Select **Finish**.

## Use PowerShell to convert the disk

You can convert a virtual disk using the [Convert-VHD](#) cmdlet in PowerShell. If you need information about installing this cmdlet see [Install the Hyper-V role](#).

### NOTE

If you're going to use Azure PowerShell to [upload your disk to Azure](#) and you have [Hyper-V](#) enabled, this step is optional. [Add-AzVHD](#) will perform it for you.

The following example converts the disk from VHDX to VHD. It also converts the disk from a dynamically expanding disk to a fixed-size disk.

```
Convert-VHD -Path C:\test\MyVM.vhdx -DestinationPath C:\test\MyNewVM.vhd -VHDTType Fixed
```

In this example, replace the value for **Path** with the path to the virtual hard disk that you want to convert. Replace the value for **DestinationPath** with the new path and name of the converted disk.

## Use Hyper-V Manager to resize the disk

### NOTE

If you're going to use Azure PowerShell to [upload your disk to Azure](#) and you have [Hyper-V](#) enabled, this step is optional. [Add-AzVHD](#) will perform it for you.

1. Open Hyper-V Manager and select your local computer on the left. In the menu above the computer list, select **Action** > **Edit Disk**.
2. On the **Locate Virtual Hard Disk** page, select your virtual disk.
3. On the **Choose Action** page, select **Expand** > **Next**.
4. On the **Locate Virtual Hard Disk** page, enter the new size in GiB > **Next**.
5. Select **Finish**.

## Use PowerShell to resize the disk

### NOTE

If you're going to use Azure PowerShell to [upload your disk to Azure](#) and you have [Hyper-V](#) enabled, this step is optional. [Add-AzVHD](#) will perform it for you.

You can resize a virtual disk using the [Resize-VHD](#) cmdlet in PowerShell. If you need information about installing this cmdlet see [Install the Hyper-V role](#).

The following example resizes the disk from 100.5 MiB to 101 MiB to meet the Azure alignment requirement.

```
Resize-VHD -Path C:\test\MyNewVM.vhd -SizeBytes 105906176
```

In this example, replace the value for **Path** with the path to the virtual hard disk that you want to resize. Replace the value for **SizeBytes** with the new size in bytes for the disk.

## Convert from VMware VMDK disk format

If you have a Windows VM image in the [VMDK file format](#), then you can use [Azure Migrate](#) to convert the VMDK and upload it to Azure.

## Complete the recommended configurations

The following settings don't affect VHD uploading. However, we strongly recommend that you configured them.

- Install the [Azure Virtual Machine Agent](#). Then you can enable VM extensions. The VM extensions implement most of the critical functionality that you might want to use with your VMs. You'll need the extensions, for example, to reset passwords or configure RDP. For more information, see the [Azure Virtual Machine Agent overview](#).
- After you create the VM in Azure, we recommend that you put the page file on the *temporal drive volume* to improve performance. You can set up the file placement as follows:

```
Set-ItemProperty -Path 'HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management' -  
Name PagingFiles -Value 'D:\pagefile.sys' -Type MultiString -Force
```

If a data disk is attached to the VM, the temporal drive volume's letter is typically *D*. This designation could be different, depending on your settings and the number of available drives.

- We recommend disabling script blockers that might be provided by antivirus software. They might interfere and block the Windows Provisioning Agent scripts executed when you deploy a new VM from your image.

## Next steps

- [Upload a Windows VM image to Azure for Resource Manager deployments](#)
- [Troubleshoot Azure Windows VM activation problems](#)

# Remove machine specific information by generalizing a VM before creating an image

9/21/2022 • 3 minutes to read • [Edit Online](#)

Generalizing a VM is not necessary for creating an image in an [Azure Compute Gallery](#) unless you specifically want to create a generalized image. Generalizing is required when creating a managed image outside of a gallery.

Generalizing removes machine specific information so the image can be used to create multiple VMs. Once the VM has been generalized, you need to let the platform know so that the boot sequence can be set correctly.

## Linux

Distribution specific instructions for preparing Linux images for Azure are available here:

- [Generic steps](#)
- [CentOS](#)
- [Debian](#)
- [Flatcar](#)
- [FreeBSD](#)
- [Oracle Linux](#)
- [OpenBSD](#)
- [Red Hat](#)
- [SUSE](#)
- [Ubuntu](#)

The following instructions only cover setting the VM to generalized. We recommend you follow the distro specific instructions for production workloads.

First you'll deprovision the VM by using the Azure VM agent to delete machine-specific files and data. Use the `waagent` command with the `-deprovision+user` parameter on your source Linux VM. For more information, see the [Azure Linux Agent user guide](#). This process can't be reversed.

1. Connect to your Linux VM with an SSH client.
2. In the SSH window, enter the following command:

```
sudo waagent -deprovision+user
```

### NOTE

Only run this command on a VM that you'll capture as an image. This command does not guarantee that the image is cleared of all sensitive information or is suitable for redistribution. The `+user` parameter also removes the last provisioned user account. To keep user account credentials in the VM, use only `-deprovision`.

3. Enter `y` to continue. You can add the `-force` parameter to avoid this confirmation step.
4. After the command completes, enter `exit` to close the SSH client. The VM will still be running at this point.

Deallocate the VM that you deprovisioned with `az vm deallocate` so that it can be generalized.

```
az vm deallocate \
--resource-group myResourceGroup \
--name myVM
```

Then the VM needs to be marked as generalized on the platform.

```
az vm generalize \
--resource-group myResourceGroup \
--name myVM
```

## Windows

Sysprep removes all your personal account and security information, and then prepares the machine to be used as an image. For information about Sysprep, see [Sysprep overview](#).

Make sure the server roles running on the machine are supported by Sysprep. For more information, see [Sysprep support for server roles](#) and [Unsupported scenarios](#).

### IMPORTANT

After you have run Sysprep on a VM, that VM is considered *generalized* and cannot be restarted. The process of generalizing a VM is not reversible. If you need to keep the original VM functioning, you should create a [copy of the VM](#) and generalize its copy.

Sysprep requires the drives to be fully decrypted. If you have enabled encryption on your VM, disable encryption before you run Sysprep.

If you plan to run Sysprep before uploading your virtual hard disk (VHD) to Azure for the first time, make sure you have [prepared your VM](#).

We do not support custom answer file in the sysprep step, hence you should not use the "/unattend:*answerfile*" switch with your sysprep command.

To generalize your Windows VM, follow these steps:

1. Sign in to your Windows VM.
2. Open a Command Prompt window as an administrator.
3. Delete the panther directory (C:\Windows\Panther).
4. Then change the directory to %windir%\system32\sysprep, and then run:

```
sysprep.exe /oobe /generalize /mode:vm /shutdown
```

5. The VM will shut down when Sysprep is finished generalizing the VM. Do not restart the VM.

**TIP**

Optional Use **DISM** to optimize your image and reduce your VM's first boot time.

To optimize your image, mount your VHD by double-clicking on it in Windows explorer, and then run DISM with the **/optimize-image** parameter.

```
DISM /image:D:\ /optimize-image /boot
```

Where D: is the mounted VHD's path.

Running **DISM /optimize-image** should be the last modification you make to your VHD. If you make any changes to your VHD prior to deployment, you'll have to run **DISM /optimize-image** again.

Once Sysprep has finished, set the status of the virtual machine to **Generalized**.

```
Set-AzVm -ResourceGroupName $rgName -Name $vmName -Generalized
```

# Upload a generalized Windows VHD and use it to create new VMs in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

This article walks you through using PowerShell to upload a VHD of a generalized VM to Azure, create an image from the VHD, and create a new VM from that image. You can upload a VHD exported from an on-premises virtualization tool or from another cloud. Using [Managed Disks](#) for the new VM simplifies the VM management and provides better availability when the VM is placed in an availability set.

For a sample script, see [Sample script to upload a VHD to Azure and create a new VM](#).

## Before you begin

- Before uploading any VHD to Azure, you should follow [Prepare a Windows VHD or VHDX to upload to Azure](#).
- Review [Plan for the migration to Managed Disks](#) before starting your migration to [Managed Disks](#).

## Generalize the source VM by using Sysprep

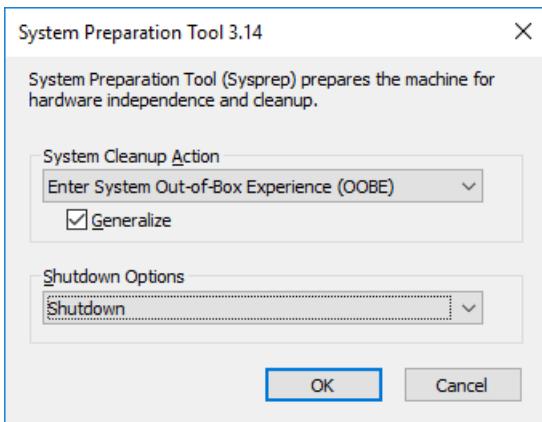
If you haven't already, you need to Sysprep the VM before uploading the VHD to Azure. Sysprep removes all your personal account information, among other things, and prepares the machine to be used as an image. For details about Sysprep, see the [Sysprep Overview](#).

Make sure the server roles running on the machine are supported by Sysprep. For more information, see [Sysprep Support for Server Roles](#).

### IMPORTANT

If you plan to run Sysprep before uploading your VHD to Azure for the first time, make sure you have [prepared your VM](#).

1. Sign in to the Windows virtual machine.
2. Open the Command Prompt window as an administrator.
3. Delete the panther directory (C:\Windows\Panther).
4. Change the directory to %windir%\system32\sysprep, and then run `sysprep.exe`.
5. In the **System Preparation Tool** dialog box, select **Enter System Out-of-Box Experience (OOBE)**, and make sure that the **Generalize** check box is enabled.
6. For **Shutdown Options**, select **Shutdown**.
7. Select **OK**.



- When Sysprep finishes, it shuts down the virtual machine. Do not restart the VM.

## Upload the VHD

You can now upload a VHD straight into a managed disk. For instructions, see [Upload a VHD to Azure using Azure PowerShell](#).

Once the VHD is uploaded to the managed disk, you need to use `Get-AzDisk` to get the managed disk.

```
$disk = Get-AzDisk -ResourceGroupName 'myResourceGroup' -DiskName 'myDiskName'
```

## Create the image

Create a managed image from your generalized OS managed disk. Replace the following values with your own information.

First, set some variables:

```
$location = 'East US'  
$imageName = 'myImage'  
$rgName = 'myResourceGroup'
```

Create the image using your managed disk.

```
$imageConfig = New-AzImageConfig `  
-Location $location  
$imageConfig = Set-AzImageOsDisk `  
-Image $imageConfig `  
-OsState Generalized `  
-OsType Windows `  
-ManagedDiskId $disk.Id
```

Create the image.

```
$image = New-AzImage `  
-ImageName $imageName `  
-ResourceGroupName $rgName `  
-Image $imageConfig
```

## Create the VM

Now that you have an image, you can create one or more new VMs from the image. This example creates a VM

named *myVM* from *myImage*, in *myResourceGroup*.

```
New-AzVm ` 
-ResourceGroupName $rgName ` 
-Name "myVM" ` 
-Image $image.Id ` 
-Location $location ` 
-VirtualNetworkName "myVnet" ` 
-SubnetName "mySubnet" ` 
-SecurityGroupName "myNSG" ` 
-PublicIpAddressName "myPIP"
```

## Next steps

Sign in to your new virtual machine. For more information, see [How to connect and log on to an Azure virtual machine running Windows](#).

# Create a managed image of a generalized VM in Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Managed images are helpful in development and test environments where you need a consistent baseline VM. A managed image resource can be created from a generalized virtual machine (VM) that is stored as either a managed disk or an unmanaged disk in a storage account. The image can then be used to create multiple VMs. For information on how managed images are billed, see [Managed Disks pricing](#).

One managed image supports up to 20 simultaneous deployments. Attempting to create more than 20 VMs concurrently, from the same managed image, may result in provisioning timeouts due to the storage performance limitations of a single VHD. To create more than 20 VMs concurrently, use an [Azure Compute Gallery](#) (formerly known as Shared Image Gallery) image configured with 1 replica for every 20 concurrent VM deployments.

## Prerequisites

You need a [generalized](#) VM in order to create an image.

## Create a managed image from a VM using the portal

1. Go to the [Azure portal](#). Search for and select **Virtual machines**.
2. Select your VM from the list.
3. In the **Virtual machine** page for the VM, on the upper menu, select **Capture**. The **Create an image** page appears.
4. For **Share image to Azure compute gallery**, select **No, capture only a managed image**.
5. For **Resource Group**, you can either create the image in the same resource group as the VM or select another resource group in your subscription.
6. For **Name**, either accept the pre-populated name or type your own name for the image.
7. If you want to delete the source VM after the image has been created, select **Automatically delete this virtual machine after creating the image**.
8. g. If you want the ability to use the image in any [availability zone](#), select **On** for **Zone resiliency**.
9. Select **Create** to create the image.

After the image is created, you can find it as an **Image** resource in the list of resources in the resource group.

## Create a managed image of a VM using PowerShell

Creating an image directly from the VM ensures that the image includes all of the disks associated with the VM, including the OS disk and any data disks. This example shows how to create a managed image from a VM that uses managed disks.

Before you begin, make sure that you have the latest version of the Azure PowerShell module. To find the version, run `Get-Module -ListAvailable Az` in PowerShell. If you need to upgrade, see [Install Azure PowerShell](#)

on Windows with [PowerShellGet](#). If you are running PowerShell locally, run `Connect-AzAccount` to create a connection with Azure.

#### NOTE

If you would like to store your image in zone-redundant storage, you need to create it in a region that supports [availability zones](#) and include the `-ZoneResilient` parameter in the image configuration (`New-AzImageConfig` command).

To create a VM image, follow these steps:

1. Create some variables.

```
$vmName = "myVM"  
$rgName = "myResourceGroup"  
$location = "EastUS"  
$imageName = "myImage"
```

2. Make sure the VM has been deallocated.

```
Stop-AzVM -ResourceGroupName $rgName -Name $vmName -Force
```

3. Set the status of the virtual machine to **Generalized**.

```
Set-AzVm -ResourceGroupName $rgName -Name $vmName -Generalized
```

4. Get the virtual machine.

```
$vm = Get-AzVM -Name $vmName -ResourceGroupName $rgName
```

5. Create the image configuration.

```
$image = New-AzImageConfig -Location $location -SourceVirtualMachineId $vm.Id
```

6. Create the image.

```
New-AzImage -Image $image -ImageName $imageName -ResourceGroupName $rgName
```

## Create an image from a managed disk using PowerShell

If you want to create an image of only the OS disk, specify the managed disk ID as the OS disk:

1. Create some variables.

```
$vmName = "myVM"  
$rgName = "myResourceGroup"  
$location = "EastUS"  
$imageName = "myImage"
```

2. Get the VM.

```
$vm = Get-AzVm -Name $vmName -ResourceGroupName $rgName
```

3. Get the ID of the managed disk.

```
$diskID = $vm.StorageProfile.OsDisk.ManagedDisk.Id
```

4. Create the image configuration.

```
$imageConfig = New-AzImageConfig -Location $location  
$imageConfig = Set-AzImageOsDisk -Image $imageConfig -OsState Generalized -OsType Windows -  
ManagedDiskId $diskID
```

5. Create the image.

```
New-AzImage -ImageName $imageName -ResourceGroupName $rgName -Image $imageConfig
```

## Create a managed image from a snapshot using PowerShell

You can create a managed image from a snapshot of a generalized VM by following these steps:

1. Create some variables.

```
$rgName = "myResourceGroup"  
$location = "EastUS"  
$snapshotName = "mySnapshot"  
$imageName = "myImage"
```

2. Get the snapshot.

```
$snapshot = Get-AzSnapshot -ResourceGroupName $rgName -SnapshotName $snapshotName
```

3. Create the image configuration.

```
$imageConfig = New-AzImageConfig -Location $location  
$imageConfig = Set-AzImageOsDisk -Image $imageConfig -OsState Generalized -OsType Windows -  
SnapshotId $snapshot.Id
```

4. Create the image.

```
New-AzImage -ImageName $imageName -ResourceGroupName $rgName -Image $imageConfig
```

## Create a managed image from a VM that uses a storage account

To create a managed image from a VM that doesn't use managed disks, you need the URI of the OS VHD in the storage account, in the following format:

<https://mystorageaccountblob.core.windows.net/vhdcontainer/vhdfilename.vhd>. In this example, the VHD is in *mystorageaccount*, in a container named *vhdcontainer*, and the VHD filename is *vhdfilename.vhd*.

1. Create some variables.

```
$vmName = "myVM"
$rgName = "myResourceGroup"
$location = "EastUS"
$imageName = "myImage"
$osVhdUri = "https://mystorageaccount.blob.core.windows.net/vhdcontainer/vhdfilename.vhd"
```

2. Stop/deallocate the VM.

```
Stop-AzVM -ResourceGroupName $rgName -Name $vmName -Force
```

3. Mark the VM as generalized.

```
Set-AzVm -ResourceGroupName $rgName -Name $vmName -Generalized
```

4. Create the image by using your generalized OS VHD.

```
$imageConfig = New-AzImageConfig -Location $location
$imageConfig = Set-AzImageOsDisk -Image $imageConfig -OsType Windows -OsState Generalized -BlobUri
$osVhdUri
$image = New-AzImage -ImageName $imageName -ResourceGroupName $rgName -Image $imageConfig
```

## Next steps

- [Create a VM from a managed image](#).
- Learn more about using an [Azure Compute Gallery](#) (formerly known as Shared Image Gallery)

# Create a VM from a managed image

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

You can create multiple virtual machines (VMs) from an Azure managed VM image using the Azure portal or PowerShell. A managed VM image contains the information necessary to create a VM, including the OS and data disks. The virtual hard disks (VHDs) that make up the image, including both the OS disks and any data disks, are stored as managed disks.

Before creating a new VM, you'll need to [create a managed VM image](#) to use as the source image and grant read access on the image to any user who should have access to the image.

One managed image supports up to 20 simultaneous deployments. Attempting to create more than 20 VMs concurrently, from the same managed image, may result in provisioning timeouts due to the storage performance limitations of a single VHD. To create more than 20 VMs concurrently, use an [Azure Compute Gallery](#) (formerly known as Shared Image Gallery) image configured with 1 replica for every 20 concurrent VM deployments.

## Use the portal

1. Go to the [Azure portal](#) to find a managed image. Search for and select **Images**.
2. Select the image you want to use from the list. The image **Overview** page opens.
3. Select **Create VM** from the menu.
4. Enter the virtual machine information. The user name and password entered here will be used to log in to the virtual machine. When complete, select **OK**. You can create the new VM in an existing resource group, or choose **Create new** to create a new resource group to store the VM.
5. Select a size for the VM. To see more sizes, select **View all** or change the **Supported disk type** filter.
6. Under **Settings**, make changes as necessary and select **OK**.
7. On the summary page, you should see your image name listed as a **Private image**. Select **Ok** to start the virtual machine deployment.

## Use PowerShell

You can use PowerShell to create a VM from an image by using the simplified parameter set for the [New-AzVm](#) cmdlet. The image needs to be in the same resource group where you'll create the VM.

The simplified parameter set for [New-AzVm](#) only requires that you provide a name, resource group, and image name to create a VM from an image. New-AzVm will use the value of the **-Name** parameter as the name of all of the resources that it creates automatically. In this example, we provide more detailed names for each of the resources but let the cmdlet create them automatically. You can also create resources beforehand, such as the virtual network, and pass the resource name into the cmdlet. New-AzVm will use the existing resources if it can find them by their name.

The following example creates a VM named *myVMFromImage*, in the *myResourceGroup* resource group, from the image named *myImage*.

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroup" ` 
    -Name "myVMfromImage" ` 
    -ImageName "myImage" ` 
    -Location "East US" ` 
    -VirtualNetworkName "myImageVnet" ` 
    -SubnetName "myImageSubnet" ` 
    -SecurityGroupName "myImageNSG" ` 
    -PublicIpAddressName "myImagePIP"
```

## Next steps

[Create and manage Windows VMs with the Azure PowerShell module](#)

# Visual Studio images on Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Using Visual Studio in a preconfigured Azure virtual machine (VM) is a quick, easy way to go from nothing to an up-and-running development environment. System images with different Visual Studio configurations are available in the [Azure Marketplace](#).

New to Azure? [Create a free Azure account](#).

## NOTE

Not all subscriptions are eligible to deploy Windows 10 images. For more information see [Use Windows client in Azure for dev/test scenarios](#)

## What configurations and versions are available?

Images for the most recent major versions, Visual Studio 2019, Visual Studio 2017 and Visual Studio 2015, can be found in the Azure Marketplace. For each released major version, you see the originally "released to web" (RTW) version and the latest updated versions. Each of these versions offers the Visual Studio Enterprise and the Visual Studio Community editions. These images are updated at least every month to include the latest Visual Studio and Windows updates. While the names of the images remain the same, each image's description includes the installed product version and the image's "as of" date.

RELEASE VERSION	EDITIONS	PRODUCT VERSION
Visual Studio 2019: Latest (Version 16.8)	Enterprise, Community	Version 16.8.0
Visual Studio 2019: RTW	Enterprise	Version 16.0.20
Visual Studio 2017: Latest (Version 15.9)	Enterprise, Community	Version 15.9.29
Visual Studio 2017: RTW	Enterprise, Community	Version 15.0.28
Visual Studio 2015: Latest (Update 3)	Enterprise, Community	Version 14.0.25431.01

## NOTE

In accordance with Microsoft servicing policy, the originally released (RTW) version of Visual Studio 2015 has expired for servicing. Visual Studio 2015 Update 3 is the only remaining version offered for the Visual Studio 2015 product line.

For more information, see the [Visual Studio Servicing Policy](#).

## What features are installed?

Each image contains the recommended feature set for that Visual Studio edition. Generally, the installation includes:

- All available workloads, including each workload's recommended optional components. More details on the workloads, components, and SDKs included Visual Studio could be found in the [Visual Studio documentation](#)
- .NET 4.6.2 and .NET 4.7 SDKs, Targeting Packs, and Developer Tools
- Visual F#
- GitHub Extension for Visual Studio
- LINQ to SQL Tools

The command line used to install Visual Studio when building the images is as follows:

```
vs_enterprise.exe --allWorkloads --includeRecommended --passive ^
    add Microsoft.Net.Component.4.7.SDK ^
    add Microsoft.Net.Component.4.7.TargetingPack ^
    add Microsoft.Net.Component.4.6.2.SDK ^
    add Microsoft.Net.Component.4.6.2.TargetingPack ^
    add Microsoft.Net.ComponentGroup.4.7.DeveloperTools ^
    add Microsoft.VisualStudio.Component.FSharp ^
    add Component.GitHub.VisualStudio ^
    add Microsoft.VisualStudio.Component.LinqToSql
```

If the images don't include a Visual Studio feature that you require, provide feedback through the feedback tool in the upper-right corner of the page.

## What size VM should I choose?

Azure offers a full range of virtual machine sizes. Because Visual Studio is a powerful, multi-threaded application, you want a VM size that includes at least two processors and 7 GB of memory. We recommend the following VM sizes for the Visual Studio images:

- Standard\_D2\_v3
- Standard\_D2s\_v3
- Standard\_D4\_v3
- Standard\_D4s\_v3
- Standard\_D2\_v2
- Standard\_D2S\_v2
- Standard\_D3\_v2

For more information on the latest machine sizes, see [Sizes for Windows virtual machines in Azure](#).

With Azure, you can rebalance your initial choice by resizing the VM. You can either provision a new VM with a more appropriate size, or resize your existing VM to different underlying hardware. For more information, see [Resize a Windows VM](#).

## After the VM is running, what's next?

Visual Studio follows the "bring your own license" model in Azure. As with an installation on proprietary hardware, one of the first steps is licensing your Visual Studio installation. To unlock Visual Studio, either:

- Sign in with a Microsoft account that's associated with a Visual Studio subscription
- Unlock Visual Studio with the product key that came with your initial purchase

For more information, see [Sign in to Visual Studio](#) and [How to unlock Visual Studio](#).

## How do I save the development VM for future or team use?

The spectrum of development environments is huge, and there's real cost associated with building out the more

complex environments. Regardless of your environment's configuration, you can save, or capture, your configured VM as a "base image" for future use or for other members of your team. Then, when booting a new VM, you provision it from the base image rather than the Azure Marketplace image.

A quick summary: Use the System Preparation tool (Sysprep) and shut down the running VM, and then capture (*Figure 1*) the VM as an image through the UI in the Azure portal. Azure saves the **.vhd** file that contains the image in the storage account of your choosing. The new image then shows up as an Image resource in your subscription's list of resources.



(*Figure 1*) Capture an image through the Azure portal UI.

For more information, see [Create a managed image of a generalized VM in Azure](#).

#### IMPORTANT

Don't forget to use Sysprep to prepare the VM. If you miss that step, Azure can't provision a VM from the image.

#### NOTE

You still incur some cost for storage of the images, but that incremental cost can be insignificant compared to the overhead costs to rebuild the VM from scratch for each team member who needs one. For instance, it costs a few dollars to create and store a 127-GB image for a month that's reusable by your entire team. However, these costs are insignificant compared to hours each employee invests to build out and validate a properly configured dev box for their individual use.

Additionally, your development tasks or technologies might need more scale, like varieties of development configurations and multiple machine configurations. You can use Azure DevTest Labs to create *recipes* that automate construction of your "golden image." You can also use DevTest Labs to manage policies for your team's running VMs. [Using Azure DevTest Labs for developers](#) is the best source for more information on DevTest Labs.

## Next steps

Now that you know about the preconfigured Visual Studio images, the next step is to create a new VM:

- [Create a VM through the Azure portal](#)
- [Windows Virtual Machines overview](#)

# Azure VM Image Builder overview

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

By using standardized virtual machine (VM) images, your organization can more easily migrate to the cloud and help ensure consistency in your deployments. Images ordinarily include predefined security, configuration settings, and any necessary software. Setting up your own imaging pipeline requires time, infrastructure, and many other details. With Azure VM Image Builder, you need only create a configuration that describes your image and submit it to the service, where the image is built and then distributed.

With VM Image Builder, you can migrate your existing image customization pipeline to Azure as you continue to use existing scripts, commands, and processes. You can integrate your core applications into a VM image, so that your VMs can take on workloads after the images are created. You can even add configurations to build images for Azure Virtual Desktop, as virtual hard discs (VHDs) for use in Azure Stack, or for ease of exporting.

VM Image Builder lets you start with Windows or Linux images either from Azure Marketplace or as existing custom images, and then add your own customizations. You can also specify where you want your resulting images to be hosted in [Azure Compute Gallery](#) (formerly Shared Image Gallery), as managed images or as VHDs.

## Features

Although it's possible to create custom VM images by hand or by other tools, the process can be cumbersome and unreliable. VM Image Builder, which is built on [HashiCorp Packer](#), gives you the benefits of a managed service.

### Simplicity

To reduce the complexity of creating VM images, VM Image Builder:

- Removes the need to use complex tooling, processes, and manual steps to create a VM image. VM Image Builder abstracts out all these details and hides Azure-specific requirements, such as the need to generalize the image (Sysprep). And it gives more advanced users the ability to override such requirements.
- Can be integrated with existing image build pipelines for a click-and-go experience. To do so, you can either call VM Image Builder from your pipeline or use an [Azure VM Image Builder service DevOps task \(preview\)](#).
- Can fetch customization data from various sources, which removes the need to collect them all from one place.
- Can be integrated with Compute Gallery, which creates an image management system with which to distribute, replicate, version, and scale images globally. Additionally, you can distribute the same resulting image as a VHD or as one or more managed images, without having to rebuild them from scratch.

### Infrastructure as code

With VM Image Builder, there's no need to manage your long-term infrastructure (for example, storage accounts that hold customization data) or transient infrastructure (for example, temporary VMs for building images).

VM Image Builder stores your VM image build artifacts as Azure resources. This feature removes both the need to maintain offline definitions and the risk of environment drifts that are caused by accidental deletions or updates.

## Security

To help keep your images secure, VM Image Builder:

- Enables you to create baseline images (that is, your minimum security and corporate configurations) and allows other departments to customize them further. You can help keep these images secure and compliant by using VM Image Builder to quickly rebuild a golden image that uses the latest patched version of a source image. VM Image Builder also makes it easier for you to build images that meet the Azure Windows security baseline. For more information, see [VM Image Builder - Windows baseline template](#).
- Enables you to fetch your customization artifacts without having to make them publicly accessible. VM Image Builder can use your [Azure Managed Identity](#) to fetch these resources, and you can restrict the privileges of this identity as tightly as required by using Azure role-based access control (Azure RBAC). You can both keep your artifacts secret and prevent tampering by unauthorized actors.
- Securely stores copies of customization artifacts, transient compute and storage resources, and their resulting images within your subscription, because access is controlled by Azure RBAC. This level of security, which also applies to the build VM that's used to create the customized image, helps prevent your customization scripts and files from being copied to an unknown VM in an unknown subscription. And you can achieve a high degree of separation from other customers' workloads by using [Isolated VM offerings](#) for the build VM.
- Enables you to connect VM Image Builder to your existing virtual networks, so that you can communicate with existing configuration servers, such as DSC (desired state configuration pull server), Chef, and Puppet, file shares, or any other routable servers and services.
- Can be configured to assign your user-assigned identities to the VM Image Builder build VM (that is, the VM that the VM Image Builder service creates in your subscription and uses to build and customize the image). You can then use these identities at customization time to access Azure resources, including secrets, in your subscription. There's no need to assign VM Image Builder direct access to those resources.

## Regions

The VM Image Builder service is available in the following regions:

### NOTE

You can still distribute images outside these regions.

- East US
- East US 2
- West Central US
- West US
- West US 2
- West US 3
- South Central US
- North Europe
- West Europe
- South East Asia
- Australia Southeast
- Australia East
- UK South

- UK West
- Brazil South
- Canada Central
- Central India
- Central US
- France Central
- Germany West Central
- Japan East
- North Central US
- Norway East
- Switzerland North
- Jio India West
- UAE North
- East Asia
- Korea Central
- South Africa North
- USGov Arizona (public preview)
- USGov Virginia (public preview)

To access the Azure VM Image Builder public preview in the Fairfax regions (USGov Arizona and USGov Virginia), you must register the *Microsoft.VirtualMachineImages/FairfaxPublicPreview* feature. To do so, run the following command:

- [Azure PowerShell](#)
- [Azure CLI](#)

```
Register-AzProviderPreviewFeature -ProviderNamespace Microsoft.VirtualMachineImages -Name FairfaxPublicPreview
```

## OS support

VM Image Builder supports the following Azure Marketplace base operating system images:

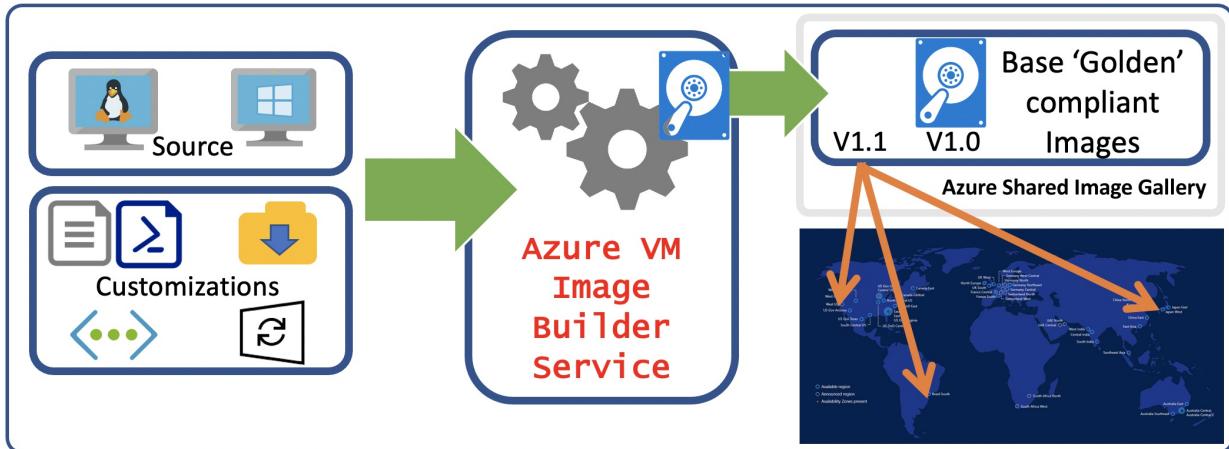
- Ubuntu 18.04
- Ubuntu 16.04
- RHEL 7.6, 7.7
- CentOS 7.6, 7.7
- SLES 12 SP4
- SLES 15, SLES 15 SP1
- Windows 10 RS5 Enterprise/Enterprise multi-session/Professional
- Windows 2016
- Windows 2019
- CBL-Mariner

### IMPORTANT

These operating systems have been tested and now work with VM Image Builder. However, VM Image Builder should work with any Linux or Windows image in the marketplace.

## How it works

VM Image Builder is a fully managed Azure service that's accessible to Azure resource providers. Resource providers configure it by specifying a source image, a customization to perform, and where the new image is to be distributed. A high-level workflow is illustrated in the following diagram:



You can pass template configurations by using Azure PowerShell, the Azure CLI, or Azure Resource Manager templates, or by using a VM Image Builder DevOps task. When you submit the configuration to the service, Azure creates an *image template resource*. When the image template resource is created, a *staging resource group* is created in your subscription, in the following format:

`IT_\<DestinationResourceGroup>_\<TemplateName>_\<(GUID)>`. The staging resource group contains files and scripts, which are referenced in the File, Shell, and PowerShell customization in the ScriptURI property.

To run the build, you invoke `Run` on the VM Image Builder template resource. The service then deploys additional resources for the build, such as a VM, network, disk, and network adapter.

If you build an image without using an existing virtual network, VM Image Builder also deploys a public IP and network security group, and it connects to the build VM by using Secure Shell (SSH) or Windows Remote Management (WinRM) protocol.

If you select an existing virtual network, the service is deployed via Azure Private Link, and a public IP address isn't required. For more information, see [VM Image Builder networking overview](#).

When the build finishes, all resources are deleted, except for the staging resource group and the storage account. You can remove them by deleting the image template resource, or you can leave them in place to run the build again.

For multiple examples, step-by-step guides, configuration templates, and solutions, go to the [VM Image Builder GitHub repository](#).

### Move support

The image template resource is immutable, and it contains links to resources and the staging resource group. Therefore, this resource type doesn't support being moved.

If you want to move the image template resource, either make sure that you have a copy of the configuration template or, if you don't have a copy, extract the existing configuration from the resource. Then, create a new image template resource in the new resource group with a new name, and delete the previous image template resource.

## Permissions

When you register for the VM Image Builder service, you're granting the service permission to create, manage, and delete a staging resource group, which is prefixed with `IT_*`. And you have rights to add to it any resources

that are required for the image build. This happens because a VM Image Builder service principal name is made available in your subscription after you've registered successfully.

To allow VM Image Builder to distribute images to either the managed images or Compute Gallery, you need to create an Azure user-assigned identity that has permissions to read and write images. If you're accessing Azure Storage, you'll need permissions to read private and public containers.

In API version 2021-10-01 and later, VM Image Builder supports adding Azure user-assigned identities to the build VM to enable scenarios where you need to authenticate with services such as Azure Key Vault in your subscription.

For more information about permissions, see

- [Configure VM Image Builder permissions by using PowerShell](#)
- [Configure VM Image Builder permissions by using the Azure CLI](#)
- [Create a VM Image Builder template](#)

## Costs

You'll incur some compute, networking, and storage costs when you create, build, and store images by using VM Image Builder. These costs are similar to those that you incur when you create custom images manually. Your resources are charged at your Azure rates.

During the image-creation process, files are downloaded and stored in the

`IT_<DestinationResourceGroup>_<TemplateName>` resource group, which incurs a small storage cost. If you don't want to keep these files, delete the image template after you've built the image.

VM Image Builder creates a VM by using the default Standard\_D1\_v2 VM size for Gen1 images and Standard\_D2ds\_v4 for Gen2 images, along with the storage and networking that's needed for the VM. These resources last for the duration of the build process and are deleted after VM Image Builder has finished creating the image.

VM Image Builder distributes the image to your chosen regions, which might incur network egress charges.

## Hyper-V generation

VM Image Builder currently supports creating Hyper-V Gen1 and Gen2 images in a Compute Gallery and as managed images or VHDs. Keep in mind that the distributed image is always in the same generation as the provided image.

For Gen2 images, ensure that you're using the correct SKU. For example, the SKU for an Ubuntu Server 18.04 Gen2 image would be 18\_04-Its-gen2. The SKU for an Ubuntu Server 18.04 Gen1 image would be 18.04-Its.

Here's how to find SKUs that are based on the image publisher:

```
# Find all Gen2 SKUs published by Microsoft Windows Desktop
az vm image list --publisher MicrosoftWindowsDesktop --sku g2 --output table --all

# Find all Gen2 SKUs published by Canonical
az vm image list --publisher Canonical --sku gen2 --output table --all
```

For more information about Azure VM images that support Gen2, see [Gen2 VM images in Azure Marketplace](#).

## Next steps

To try out VM Image Builder, see the articles about building [Linux](#) or [Windows](#) images.

# Create a Linux image and distribute it to an Azure Compute Gallery by using the Azure CLI

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In this article, you learn how to use Azure VM Image Builder and the Azure CLI to create an image version in an [Azure Compute Gallery](#) (formerly Shared Image Gallery) and then distribute the image globally. You can also create an image version by using [Azure PowerShell](#).

This article uses a sample JSON template to configure the image. The JSON file is at [helloImageTemplateforSIG.json](#).

To distribute the image to an Azure Compute Gallery, the template uses `sharedImage` as the value for the `distribute` section of the template.

## Register the features

To use VM Image Builder, you need to register the feature. Check your registration by running the following commands:

```
az provider show -n Microsoft.VirtualMachineImages -o json | grep registrationState  
az provider show -n Microsoft.KeyVault -o json | grep registrationState  
az provider show -n Microsoft.Compute -o json | grep registrationState  
az provider show -n Microsoft.Storage -o json | grep registrationState  
az provider show -n Microsoft.Network -o json | grep registrationState
```

If the output doesn't say *registered*, run the following commands:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Set variables and permissions

Because you'll be using some pieces of information repeatedly, create some variables to store that information.

VM Image Builder supports creating custom images only in the same resource group as the source-managed image. In the following example, update the resource group name to be the same resource group as your source-managed image.

```
# Resource group name - ibLinuxGalleryRG in this example
sigResourceGroup=ibLinuxGalleryRG
# Datacenter location - West US 2 in this example
location=westus2
# Additional region to replicate the image to - East US in this example
additionalRegion=eastus
# Name of the Azure Compute Gallery - myGallery in this example
sigName=myIbGallery
# Name of the image definition to be created - myImageDef in this example
imageDefName=myIbImageDef
# Reference name in the image distribution metadata
runOutputName=aibLinuxSIG
```

Create a variable for your subscription ID:

```
subscriptionID=$(az account show --query id --output tsv)
```

Create the resource group:

```
az group create -n $sigResourceGroup -l $location
```

## Create a user-assigned identity and set permissions on the resource group

VM Image Builder uses the provided [user-identity](#) to inject the image into an Azure Compute Gallery. In this example, you create an Azure role definition with specific actions for distributing the image. The role definition is then assigned to the user identity.

```

# Create user-assigned identity for VM Image Builder to access the storage account where the script is stored
identityName=aibBuiUserId$(date +'%s')
az identity create -g $sigResourceGroup -n $identityName

# Get the identity ID
imgBuilderCliId=$(az identity show -g $sigResourceGroup -n $identityName --query clientId -o tsv)

# Get the user identity URI that's needed for the template
imgBuilderId=/subscriptions/$subscriptionID/resourcegroups/$sigResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/$identityName

# Download an Azure role-definition template, and update the template with the parameters that were specified earlier
curl
https://raw.githubusercontent.com/Azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleImageCreation.json -o aibRoleImageCreation.json

imageRoleDefName="Azure Image Builder Image Def"$(date +'%s')

# Update the definition
sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/$sigResourceGroup/g" aibRoleImageCreation.json
sed -i -e "s/Azure Image Builder Service Image Creation Role/$imageRoleDefName/g" aibRoleImageCreation.json

# Create role definitions
az role definition create --role-definition ./aibRoleImageCreation.json

# Grant a role definition to the user-assigned identity
az role assignment create \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup

```

## Create an image definition and gallery

To use VM Image Builder with Azure Compute Gallery, you need to have an existing gallery and image definition. VM Image Builder doesn't create the gallery and image definition for you.

If you don't already have a gallery and image definition to use, start by creating them.

First, create a gallery:

```

az sig create \
-g $sigResourceGroup \
--gallery-name $sigName

```

Then, create an image definition:

```

az sig image-definition create \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--publisher myIbPublisher \
--offer myOffer \
--sku 18.04-LTS \
--os-type Linux

```

## Download and configure the JSON file

Download the JSON template and configure it with your variables:

```
curl  
https://raw.githubusercontent.com/Azure/azvmimagebuilder/master/quickstarts/1_Creating_a_Custom_Linux_Shared_Image_Gallery_Image/helloImageTemplateforSIG.json -o helloImageTemplateforSIG.json  
sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateforSIG.json  
sed -i -e "s/<rgName>/$sigResourceGroup/g" helloImageTemplateforSIG.json  
sed -i -e "s/<imageDefName>/$imageDefName/g" helloImageTemplateforSIG.json  
sed -i -e "s/<sharedImageGalName>/$sigName/g" helloImageTemplateforSIG.json  
sed -i -e "s/<region1>/$location/g" helloImageTemplateforSIG.json  
sed -i -e "s/<region2>/$additionalRegion/g" helloImageTemplateforSIG.json  
sed -i -e "s/<runOutputName>/$runOutputName/g" helloImageTemplateforSIG.json  
sed -i -e "s%<imgBuilderId>%$imgBuilderId%" helloImageTemplateforSIG.json
```

## Create the image version

In this section you create the image version in the gallery.

Submit the image configuration to the Azure VM Image Builder service:

```
az resource create \  
    --resource-group $sigResourceGroup \  
    --properties @helloImageTemplateforSIG.json \  
    --is-full-object \  
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \  
    -n helloImageTemplateforSIG01
```

Start the image build:

```
az resource invoke-action \  
    --resource-group $sigResourceGroup \  
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \  
    -n helloImageTemplateforSIG01 \  
    --action Run
```

It can take a few moments to create the image and replicate it to both regions. Wait until this part is finished before you move on to create a VM.

## Create the VM

Create the VM from the image version that was created by VM Image Builder.

```
az vm create \  
    --resource-group $sigResourceGroup \  
    --name myAibGalleryVM \  
    --admin-username aibuser \  
    --location $location \  
    --image  
    "/subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup/providers/Microsoft.Compute/galleries/$sigName/images/$imageDefName/versions/latest" \  
    --generate-ssh-keys
```

Connect to the VM via Secure Shell (SSH):

```
ssh aibuser@<publicIpAddress>
```

As soon as your SSH connection is established, you should see that the image was customized with a *Message*

of the Day.

```
*****
** This VM was built from the:      **
** !! AZURE VM IMAGE BUILDER Custom Image !!   **
** You have just been Customized :-)      **
*****
```

## Clean up your resources

### NOTE

If you now want to try to recustomize the image version to create a new version of the same image, *skip the step outlined here* and go to [Use VM Image Builder to create another image version](#).

If you no longer need the resources that were created as you followed the process in this article, you can delete them by doing the following.

This process deletes both the image that you created and all the other resource files. Make sure that you've finished this deployment before you delete the resources.

When you're deleting gallery resources, you need to delete all the image versions before you can delete the image definition that was used to create them. To delete a gallery, you first need to have deleted all the image definitions in the gallery.

1. Delete the VM Image Builder template.

```
az resource delete \
--resource-group $sigResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateforSIG01
```

2. Delete permissions assignments, roles, and identity.

```
az role assignment delete \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup

az role definition delete --name "$imageRoleDefName"

az identity delete --ids $imgBuilderId
```

3. Get the image version that was created by VM Image Builder (it always starts with `0.`), and then delete it.

```
sigDefImgVersion=$(az sig image-version list \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID --query [].'name' -o json | grep 0. | tr -d ''')
az sig image-version delete \
-g $sigResourceGroup \
--gallery-image-version $sigDefImgVersion \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID
```

4. Delete the image definition.

```
az sig image-definition delete \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID
```

5. Delete the gallery.

```
az sig delete -r $sigName -g $sigResourceGroup
```

6. Delete the resource group.

```
az group delete -n $sigResourceGroup -y
```

## Next steps

Learn more about [Azure Compute Gallery](#).

# Create a Windows VM by using Azure VM Image Builder

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this article, you learn how to create a customized Windows image by using Azure VM Image Builder. The example in this article uses [customizers](#) for customizing the image:

- PowerShell (ScriptUri): Download and run a [PowerShell script](#).
- Windows Restart: Restarts the VM.
- PowerShell (inline): Runs a specific command. In this example, it creates a directory on the VM by using `mkdir c:\\buildActions`.
- File: Copies a file from GitHub to the VM. This example copies `index.md` to `c:\\buildArtifacts\\index.html` on the VM.
- `buildTimeoutInMinutes` : Specifies a build time, in minutes. The default is 240 minutes, which you can increase to allow for longer-running builds. The minimum allowed value is 6 minutes. Values shorter than 6 minutes will cause errors.
- `vmProfile` : Specifies a `vmSize` and network properties.
- `osDiskSizeGB` : Can be used to increase the size of an image.
- `identity` . Provides an identity for VM Image Builder to use during the build.

Use the following sample JSON template to configure the image: [helloImageTemplateWin.json](#).

## NOTE

Windows users can run the following Azure CLI examples on [Azure Cloud Shell](#) by using Bash.

## Register the features

To use VM Image Builder, you need to register the feature. Check your registration by running the following commands:

```
az provider show -n Microsoft.VirtualMachineImages | grep registrationState  
az provider show -n Microsoft.KeyVault | grep registrationState  
az provider show -n Microsoft.Compute | grep registrationState  
az provider show -n Microsoft.Storage | grep registrationState  
az provider show -n Microsoft.Network | grep registrationState
```

If the output doesn't say *registered*, run the following commands:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Set variables

Because you'll be using some pieces of information repeatedly, create some variables to store that information:

```
# Resource group name - we're using myImageBuilderRG in this example
imageResourceGroup='myWinImgBuilderRG'
# Region location
location='WestUS2'
# Run output name
runOutputName='aibWindows'
# The name of the image to be created
imageName='aibWinImage'
```

Create a variable for your subscription ID:

```
subscriptionID=$(az account show --query id --output tsv)
```

## Create the resource group

To store the image configuration template artifact and the image, use the following resource group:

```
az group create -n $imageResourceGroup -l $location
```

## Create a user-assigned identity and set permissions on the resource group

VM Image Builder uses the provided [user-identity](#) to inject the image into the resource group. In this example, you create an Azure role definition with specific permissions for distributing the image. The role definition is then assigned to the user identity.

## Create a user-assigned managed identity and grant permissions

Create a user-assigned identity so that VM Image Builder can access the storage account where the script is stored.

```

identityName=aibBuiUserId$(date +'%s')
az identity create -g $imageResourceGroup -n $identityName

# Get the identity ID
imgBuilderCliId=$(az identity show -g $imageResourceGroup -n $identityName --query clientId -o tsv)

# Get the user identity URI that's needed for the template
imgBuilderId=/subscriptions/$subscriptionID/resourcegroups/$imageResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/$identityName

# Download the preconfigured role definition example
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aib
RoleImageCreation.json -o aibRoleImageCreation.json

imageRoleDefName="Azure Image Builder Image Def"$(date +'%s')

# Update the definition
sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" aibRoleImageCreation.json
sed -i -e "s/Azure Image Builder Service Image Creation Role/$imageRoleDefName/g" aibRoleImageCreation.json

# Create role definitions
az role definition create --role-definition ./aibRoleImageCreation.json

# Grant a role definition to the user-assigned identity
az role assignment create \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup

```

## Download the image configuration template

We've created a parameterized image configuration template for you to try. Download the example JSON file, and then configure it with the variables that you set earlier.

```

curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/0_Creating_a_Custom_Windows
_Managed_Image/helloImageTemplateWin.json -o helloImageTemplateWin.json

sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateWin.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" helloImageTemplateWin.json
sed -i -e "s/<region>/$location/g" helloImageTemplateWin.json
sed -i -e "s/<imageName>/$imageName/g" helloImageTemplateWin.json
sed -i -e "s/<runOutputName>/$runOutputName/g" helloImageTemplateWin.json
sed -i -e "s/<imgBuilderId>/$imgBuilderId/g" helloImageTemplateWin.json

```

You can modify this example in the terminal by using a text editor such as `vi`.

```
vi helloImageTemplateWin.json
```

### NOTE

For the source image, always [specify a version](#). You can't specify `latest` as the version.

If you add or change the resource group that the image is distributed to, make sure that the [permissions are set](#) on the resource group.

## Create the image

Submit the image configuration to the VM Image Builder service by running the following commands:

```
az resource create \
--resource-group $imageResourceGroup \
--properties @helloImageTemplateWin.json \
--is-full-object \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateWin01
```

When you're done, a success message is returned to the console, and a VM Image Builder configuration template is created in the `$imageResourceGroup`. To view this resource in the resource group, go to the Azure portal, and then enable **Show hidden types**.

In the background, VM Image Builder also creates a staging resource group in your subscription. This resource group is used to build the image in the following format: `IT_<DestinationResourceGroup>_<TemplateName>`.

### NOTE

Don't delete the staging resource group directly. First, delete the image template artifact, which causes the staging resource group to be deleted.

If the service reports a failure when you submit the image configuration template, do the following:

- See [Troubleshoot the Azure VM Image Builder service](#).
- Before you try to resubmit the template, delete it by running the following commands:

```
az resource delete \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateWin01
```

## Start the image build

Start the image-building process by using `az resource invoke-action`.

```
az resource invoke-action \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateWin01 \
--action Run
```

Wait until the build is complete.

If you encounter any errors, see [Troubleshoot the Azure VM Image Builder service](#).

## Create the VM

Create the VM by using the image that you built. In the following code, replace `<password>` with your own password for the `aibuser` on the VM.

```
az vm create \
--resource-group $imageResourceGroup \
--name aibImgWinVm00 \
--admin-username aibususer \
--admin-password <password> \
--image $imageName \
--location $location
```

## Verify the customization

Create a Remote Desktop connection to the VM by using the username and password that you set when you created the VM. In the VM, open a Command Prompt window, and then type:

```
dir c:\
```

The following two directories are created during the image customization:

- buildActions
- buildArtifacts

## Clean up your resources

When you're done, delete the resources you've created.

1. Delete the VM Image Builder template.

```
az resource delete \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateWin01
```

2. Delete the role assignment, role definition, and user identity.

```
az role assignment delete \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup

az role definition delete --name "$imageRoleDefName"

az identity delete --ids $imgBuilderId
```

3. Delete the image resource group.

```
az group delete -n $imageResourceGroup
```

## Next steps

To learn more about the components of the JSON file that this article uses, see the [VM Image Builder template reference](#).

# Create a Windows VM with VM Image Builder by using PowerShell

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article demonstrates how to create a customized Windows VM image by using the Azure VM Image Builder PowerShell module.

## Prerequisites

If you don't have an Azure subscription, [create a free account](#) before you begin.

If you choose to use PowerShell locally, this article requires that you install the Azure PowerShell module and connect to your Azure account by using the [Connect-AzAccount](#) cmdlet. For more information, see [Install Azure PowerShell](#).

## Azure Cloud Shell

Azure hosts Azure Cloud Shell, an interactive shell environment that you can use through your browser. You can use either Bash or PowerShell with Cloud Shell to work with Azure services. You can use the Cloud Shell preinstalled commands to run the code in this article, without having to install anything on your local environment.

To start Azure Cloud Shell:

OPTION	EXAMPLE/LINK
Select <b>Try It</b> in the upper-right corner of a code or command block. Selecting <b>Try It</b> doesn't automatically copy the code or command to Cloud Shell.	
Go to <a href="https://shell.azure.com">https://shell.azure.com</a> , or select the <b>Launch Cloud Shell</b> button to open Cloud Shell in your browser.	
Select the <b>Cloud Shell</b> button on the menu bar at the upper right in the <a href="#">Azure portal</a> .	

To use Azure Cloud Shell:

1. Start Cloud Shell.
2. Select the **Copy** button on a code block (or command block) to copy the code or command.
3. Paste the code or command into the Cloud Shell session by selecting **Ctrl+Shift+V** on Windows and Linux, or by selecting **Cmd+Shift+V** on macOS.
4. Select **Enter** to run the code or command.

If you have multiple Azure subscriptions, choose the appropriate subscription in which the resources should be billed. Select a specific subscription by using the [Set-AzContext](#) cmdlet.

```
Set-AzContext -SubscriptionId 00000000-0000-0000-0000-000000000000
```

## Register features

If you haven't already done so, register the following resource providers to use with your Azure subscription:

- Microsoft.Compute
- Microsoft.KeyVault
- Microsoft.Storage
- Microsoft.Network
- Microsoft.VirtualMachineImages

```
Get-AzResourceProvider -ProviderNamespace Microsoft.Compute, Microsoft.KeyVault, Microsoft.Storage,  
Microsoft.VirtualMachineImages, Microsoft.Network |  
Where-Object RegistrationState -ne Registered |  
Register-AzResourceProvider
```

## Define variables

Because you'll be using some pieces of information repeatedly, create some variables to store that information:

```
# Destination image resource group name  
$imageResourceGroup = 'myWinImgBuilderRG'  
  
# Azure region  
$location = 'WestUS2'  
  
# Name of the image to be created  
$imageTemplateName = 'myWinImage'  
  
# Distribution properties of the managed image upon completion  
$runOutputName = 'myDistResults'
```

Create a variable for your Azure subscription ID. To confirm that the `subscriptionID` variable contains your subscription ID, you can run the second line in the following example:

```
# Your Azure Subscription ID  
$subscriptionID = (Get-AzContext).Subscription.Id  
Write-Output $subscriptionID
```

## Create a resource group

Create an [Azure resource group](#) by using the `New-AzResourceGroup` cmdlet. A resource group is a logical container in which Azure resources are deployed and managed as a group.

The following example creates a resource group that's based on the name in the `$imageResourceGroup` variable in the region that you've specified in the `$location` variable. This resource group is used to store the image configuration template artifact and the image.

```
New-AzResourceGroup -Name $imageResourceGroup -Location $location
```

## Create a user identity and set role permissions

Grant Azure image builder permissions to create images in the specified resource group by using the following example. Without this permission, the image build process won't finish successfully.

1. Create variables for the role definition and identity names. These values must be unique.

```
[int]$timeInt = $(Get-Date -UFormat '%s')
$imageRoleDefName = "Azure Image Builder Image Def $timeInt"
$identityName = "myIdentity$timeInt"
```

2. Create a user identity.

```
New-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName
```

3. Store the identity resource and principal IDs in variables.

```
$identityNameResourceId = (Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName).Id
$identityNamePrincipalId = (Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName).PrincipalId
```

## Assign permissions for the identity to distribute the images

1. Download the JSON configuration file, and then modify it based on the settings that are defined in this article.

```
$myRoleImageCreationUrl =
'https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleImageCreation.json'
$myRoleImageCreationPath = "$env:TEMP\myRoleImageCreation.json"

Invoke-WebRequest -Uri $myRoleImageCreationUrl -OutFile $myRoleImageCreationPath -UseBasicParsing

$content = Get-Content -Path $myRoleImageCreationPath -Raw
$content = $content -replace '<subscriptionID>', $subscriptionID
$content = $content -replace '<rgName>', $imageResourceGroup
$content = $content -replace 'Azure Image Builder Service Image Creation Role', $imageRoleDefName
$content | Out-File -FilePath $myRoleImageCreationPath -Force
```

2. Create the role definition.

```
New-AzRoleDefinition -InputFile $myRoleImageCreationPath
```

3. Grant the role definition to the VM Image Builder service principal.

```
$RoleAssignParams = @{
    ObjectId = $identityNamePrincipalId
    RoleDefinitionName = $imageRoleDefName
    Scope = "/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"
}
New-AzRoleAssignment @RoleAssignParams
```

### NOTE

If you receive the error "New-AzRoleDefinition: Role definition limit exceeded. No more role definitions can be created," see [Troubleshoot Azure RBAC \(role-based access control\)](#).

# Create an Azure Compute Gallery

1. Create the gallery.

```
$myGalleryName = 'myImageGallery'  
$imageDefName = 'winSvrImages'  
  
New-AzGallery -GalleryName $myGalleryName -ResourceGroupName $imageResourceGroup -Location $location
```

2. Create a gallery definition.

```
$GalleryParams = @{  
    GalleryName = $myGalleryName  
    ResourceGroupName = $imageResourceGroup  
    Location = $location  
    Name = $imageDefName  
    OsState = 'generalized'  
    OsType = 'Windows'  
    Publisher = 'myCo'  
    Offer = 'Windows'  
    Sku = 'Win2019'  
}  
New-AzGalleryImageDefinition @GalleryParams
```

# Create an image

1. Create a VM Image Builder source object. For valid parameter values, see [Find Windows VM images in Azure Marketplace with Azure PowerShell](#).

```
$SrcObjParams = @{  
    SourceTypePlatformImage = $true  
    Publisher = 'MicrosoftWindowsServer'  
    Offer = 'WindowsServer'  
    Sku = '2019-Datacenter'  
    Version = 'latest'  
}  
$srcPlatform = New-AzImageBuilderTemplateSourceObject @SrcObjParams
```

2. Create a VM Image Builder distributor object.

```
$disObjParams = @{  
    SharedImageDistributor = $true  
    ArtifactTag = @{$tag='dis-share'}  
    GalleryImageId =  
        "/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup/providers/Microsoft.Compute/galleries/$myGalleryName/images/$imageDefName"  
    ReplicationRegion = $location  
    RunOutputName = $runOutputName  
    ExcludeFromLatest = $false  
}  
$disSharedImg = New-AzImageBuilderTemplateDistributorObject @disObjParams
```

3. Create a VM Image Builder customization object.

```
$ImgCustomParams01 = @{
    PowerShellCustomizer = $true
    CustomizerName = 'settingUpMgmtAgtPath'
    RunElevated = $false
    Inline = @("mkdir c:\\buildActions", "mkdir c:\\buildArtifacts", "echo Azure-Image-Builder-Was-Here > c:\\buildActions\\buildActionsOutput.txt")
}
$Customizer01 = New-AzImageBuilderTemplateCustomizerObject @ImgCustomParams01
```

4. Create a second VM Image Builder customization object.

```
$ImgCustomParams02 = @{
    FileCustomizer = $true
    CustomizerName = 'downloadBuildArtifacts'
    Destination = 'c:\\buildArtifacts\\index.html'
    SourceUri =
'https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/exampleArtifacts/buildArtifacts/index.html'
}
$Customizer02 = New-AzImageBuilderTemplateCustomizerObject @ImgCustomParams02
```

5. Create a VM Image Builder template.

```
$ImgTemplateParams = @{
    ImageTemplateName = $imageName
    ResourceGroupName = $imageResourceGroup
    Source = $srcPlatform
    Distribute = $disSharedImg
    Customize = $Customizer01, $Customizer02
    Location = $location
    UserAssignedIdentityId = $identityNameResourceId
}
New-AzImageBuilderTemplate @ImgTemplateParams
```

When the template has been created, a message is returned, and a VM Image Builder configuration template is created in `$imageResourceGroup`.

To determine whether the template creation process was successful, use the following example:

```
Get-AzImageBuilderTemplate -ImageTemplateName $imageName -ResourceGroupName $imageResourceGroup |
Select-Object -Property Name, LastRunStatusRunState, LastRunStatusMessage, ProvisioningState
```

In the background, VM Image Builder also creates a staging resource group in your subscription. This resource group is used for the image build. It's in the format `IT_<DestinationResourceGroup>_<TemplateName>`.

**WARNING**

Don't delete the staging resource group directly. To cause the staging resource group to be deleted, delete the image template artifact.

If the service reports a failure when the image configuration template is submitted, do the following:

- See [Troubleshoot Azure VM Image Builder failures](#).
- Before you retry submitting the template, delete it by following this example:

```
Remove-AzImageBuilderTemplate -ImageTemplateName $imageTemplateName -ResourceGroupName  
$imageResourceGroup
```

## Start the image build

Submit the image configuration to the VM Image Builder service by running the following command:

```
Start-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name $imageTemplateName
```

Wait for the image building process to finish, which could take up to an hour.

If you encounter errors, review [Troubleshoot Azure VM Image Builder failures](#).

## Create a VM

1. Store the VM login credentials in a variable. The password must be complex.

```
$Cred = Get-Credential
```

2. Create the VM by using the image you created.

```
$ArtifactId = (Get-AzImageBuilderTemplateRunOutput -ImageTemplateName $imageTemplateName -  
ResourceGroupName $imageResourceGroup).ArtifactId  
  
New-AzVM -ResourceGroupName $imageResourceGroup -Image $ArtifactId -Name myWinVM01 -Credential $Cred
```

## Verify the customizations

1. Create a Remote Desktop connection to the VM by using the username and password that you set when you created the VM.
2. Inside the VM, open PowerShell and run `Get-Content`, as shown in the following example:

```
Get-Content -Path C:\buildActions\buildActionsOutput.txt
```

The output is based on the contents of the file that you created during the image customization process.

```
Azure-Image-Builder-Was-Here
```

3. From the same PowerShell session, verify that the second customization finished successfully by checking for the presence of `c:\buildArtifacts\index.html`, as shown in the following example:

```
Get-ChildItem c:\buildArtifacts\
```

The result should be a directory listing showing that the file was downloaded during the image customization process.

Directory: C:\buildArtifacts			
Mode	LastWriteTime	Length	Name
-a---	29/01/2021 10:04	276	index.html

## Clean up your resources

If you no longer need the resources that were created during this process, you can delete them by doing the following:

1. Delete the VM Image Builder template.

```
Remove-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name $imageTemplateName
```

2. Delete the image resource group.

**Caution**

The following example deletes the specified resource group and all the resources that it contains. If any resources outside the scope of this article exist in the resource group, they'll also be deleted.

```
Remove-AzResourceGroup -Name $imageResourceGroup
```

## Next steps

To learn more about the components of the JSON file that this article uses, see the [VM Image Builder template reference](#).

# Create an Azure Virtual Desktop image by using VM Image Builder and PowerShell

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this article, you learn how to create an Azure Virtual Desktop image with these customizations:

- [FSLogix setup](#)
- [Azure Virtual Desktop optimization](#)
- [Microsoft Teams installation](#)
- [Windows Restart customizer](#)
- [Windows Update customizer](#)

The article discusses how to automate the customizations by using Azure VM Image Builder. You can then distribute the image to an [Azure Compute Gallery](#) (formerly Shared Image Gallery), where you can replicate it to other regions, control the scale, and share the image within and beyond your organization.

To simplify deploying a VM Image Builder configuration, our example uses an Azure Resource Manager template with the VM Image Builder template nested within it. This approach gives you a few more benefits, such as variables and parameter inputs. You can also pass parameters from the command line.

This article is intended as a copy-and-paste exercise.

## NOTE

You'll find the scripts for installing the apps on [GitHub](#). They're for illustration and testing purposes only. Do not use them for production workloads.

## Tips for building Windows images

- VM size: For Windows, use `Standard_D2_v2` or greater. The default size is `Standard_D1_v2`, which isn't suitable for Windows.
- This article uses [PowerShell customizer scripts](#). Use the following settings, or the build will stop responding:

```
"runElevated": true,  
"runAsSystem": true,
```

For example:

```
{  
    "type": "PowerShell",  
    "name": "installFSLogix",  
    "runElevated": true,  
    "runAsSystem": true,  
    "scriptUri":  
        "https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/14_Building_Images_WVD/0_i  
nstallConfFSLogix.ps1"
```

- Comment your code: The VM Image Builder build log, *customization.log*, is verbose. If you comment your scripts by using 'write-host', they'll be sent to the logs, which should make troubleshooting easier.

```
write-host 'AIB Customization: Starting OS Optimizations script'
```

- Exit codes: VM Image Builder expects all scripts to return a `0` exit code. If you use a non-zero exit code, VM Image Builder fails the customization and stops the build. If you have complex scripts, add instrumentation and emit exit codes, which will be shown in the *customization.log* file.

```
Write-Host "Exit code: " $LASTEXITCODE
```

- Test: Test and retest your code on a standalone VM. Ensure that there are no user prompts, that you're using the correct privileges, and so on.
- Networking: `Set-NetAdapterAdvancedProperty` is set in the optimization script but fails the VM Image Builder build. Because it disconnects the network, it's commented out. We're investigating this issue.

## Prerequisites

You must have the latest Azure PowerShell cmdlets installed. For more information, see [Overview of Azure PowerShell](#).

```
# Check to ensure that you're registered for the providers and RegistrationState is set to 'Registered'
Get-AzResourceProvider -ProviderNamespace Microsoft.VirtualMachineImages
Get-AzResourceProvider -ProviderNamespace Microsoft.Storage
Get-AzResourceProvider -ProviderNamespace Microsoft.Compute
Get-AzResourceProvider -ProviderNamespace Microsoft.KeyVault

# If they don't show as 'Registered', run the the following commented-out code

## Register-AzResourceProvider -ProviderNamespace Microsoft.VirtualMachineImages
## Register-AzResourceProvider -ProviderNamespace Microsoft.Storage
## Register-AzResourceProvider -ProviderNamespace Microsoft.Compute
## Register-AzResourceProvider -ProviderNamespace Microsoft.KeyVault
```

## Set up the environment and variables

```

# Step 1: Import module
Import-Module Az.Accounts

# Step 2: get existing context
$currentAzContext = Get-AzContext

# Destination image resource group
$imageResourceGroup="avdImageDemoRg"

# Location (see possible locations in the main docs)
$location="westus2"

# Your subscription. This command gets your current subscription
$subscriptionID=$currentAzContext.Subscription.Id

# Image template name
$imageTemplateName="avd10ImageTemplate01"

# Distribution properties object name (runOutput). Gives you the properties of the managed image on
completion
$runOutputName="sigOutput"

# Create resource group
New-AzResourceGroup -Name $imageResourceGroup -Location $location

```

## Permissions, user identity, and role

1. Create a user identity.

```

# setup role def names, these need to be unique
$timeInt=$(get-date -UFormat "%s")
$imageRoleDefName="Azure Image Builder Image Def"+$timeInt
$identityName="aibIdentity"+$timeInt

## Add Azure PowerShell modules to support AzUserAssignedIdentity and Azure VM Image Builder
'Az.ImageBuilder', 'Az.ManagedServiceIdentity' | ForEach-Object {Install-Module -Name $_ -AllowPrerelease}

# Create the identity
New-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName

$identityNameResourceId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).Id
$identityNamePrincipalId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).PrincipalId

```

2. Assign permissions to the identity to distribute images. The following commands download and update the template with the previously specified parameters.

```

$aibRoleImageCreationUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/1
2_Creating_AIB_Security_Roles/aibRoleImageCreation.json"
$aibRoleImageCreationPath = "aibRoleImageCreation.json"

# Download the config
Invoke-WebRequest -Uri $aibRoleImageCreationUrl -OutFile $aibRoleImageCreationPath -UseBasicParsing

((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<subscriptionID>', $subscriptionID) |
Set-Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<rgName>', $imageResourceGroup) | Set-
Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace 'Azure Image Builder Service Image
Creation Role', $imageRoleDefName) | Set-Content -Path $aibRoleImageCreationPath

# Create a role definition
New-AzRoleDefinition -InputFile ./aibRoleImageCreation.json

# Grant the role definition to the VM Image Builder service principal
New-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $imageRoleDefName -Scope
"/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"

```

#### NOTE

If you receive the error "New-AzRoleDefinition: Role definition limit exceeded. No more role definitions can be created," see [Troubleshoot Azure RBAC \(role-based access control\)](#).

## Create an Azure Compute Gallery

If you don't already have an Azure Compute Gallery, you need to create one.

```

$sigGalleryName= "myaibsig01"
$imageDefName ="win10avd"

# Create the gallery
New-AzGallery -GalleryName $sigGalleryName -ResourceGroupName $imageResourceGroup -Location $location

# Create the gallery definition
New-AzGalleryImageDefinition -GalleryName $sigGalleryName -ResourceGroupName $imageResourceGroup -Location
$location -Name $imageDefName -OsState generalized -OsType Windows -Publisher 'myCo' -Offer 'Windows' -Sku
'10avd'

```

## Configure the VM Image Builder template

For this example, we've prepared a template that downloads and updates the VM Image Builder template with the parameters that were specified earlier. The template installs FSLogix, operating system optimizations, and Microsoft Teams, and it runs Windows Update at the end.

If you open the template, you can see in the source property the image that's being used. In this example, it uses a Windows 10 multi-session image.

### Windows 10 images

You should be aware of two key types of images: multi-session and single-session.

Multi-session images are intended for pooled usage. Here's an example of the image details in Azure:

```
"publisher": "MicrosoftWindowsDesktop",
"offer": "Windows-10",
"sku": "20h2-evd",
"version": "latest"
```

Single-session images are intended for individual usage. Here's an example of the image details in Azure:

```
"publisher": "MicrosoftWindowsDesktop",
"offer": "Windows-10",
"sku": "19h2-ent",
"version": "latest"
```

You can also change which Windows 10 images are available:

```
Get-AzVMImageSku -Location westus2 -PublisherName MicrosoftWindowsDesktop -Offer windows-10
```

## Download and configure the template

Now, download the template and configure it for your own use.

```
$templateUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/14_Building_Images_W
VD/armTemplateWVD.json"
$templateFilePath = "armTemplateWVD.json"

Invoke-WebRequest -Uri $templateUrl -OutFile $templateFilePath -UseBasicParsing

((Get-Content -path $templateFilePath -Raw) -replace '<subscriptionID>',$subscriptionID) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<rgName>',$imageResourceGroup) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<region>',$location) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<runOutputName>',$runOutputName) | Set-Content -Path
$templateFilePath

((Get-Content -path $templateFilePath -Raw) -replace '<imageDefName>',$imageDefName) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<sharedImageGalName>',$sigGalleryName) | Set-Content -
Path $templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<region1>',$location) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<imgBuilderId>',$identityNameResourceId) | Set-Content
-Path $templateFilePath
```

Feel free to view the [template](#). All the code is viewable.

## Submit the template

Your template must be submitted to the service. Doing so downloads any dependent artifacts, such as scripts, and validates, checks permissions, and stores them in the staging resource group, which is prefixed with `/T_`.

```
New-AzResourceGroupDeployment -ResourceGroupName $imageResourceGroup -TemplateFile $templateFilePath -  
TemplateParameterObject @{"api-Version" = "2020-02-14"} -imageName $imageTemplateName -svclocation  
$location  
  
# Optional - if you have any errors running the preceding command, run:  
$getStatus=$(Get-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name $imageTemplateName)  
$getStatus.ProvisioningErrorCode  
$getStatus.ProvisioningErrorMessage
```

## Build the image

```
Start-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name $imageTemplateName -NoWait
```

### NOTE

The command doesn't wait for the VM Image Builder service to complete the image build, so you can query the status as shown here.

```
$getStatus=$(Get-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name $imageTemplateName)  
  
# Shows all the properties  
$getStatus | Format-List -Property *  
  
# Shows the status of the build  
$getStatus.LastRunStatusRunState  
$getStatus.LastRunStatusMessage  
$getStatus.LastRunStatusRunSubState
```

## Create a VM

Now that the image is built, you can build a VM from it. Use the examples from [New-AzVM \(Az PowerShell module.Compute\)](#).

## Clean up your resources

If you no longer need the resources that were created during this process, you can delete them by doing the following:

### IMPORTANT

Delete the resource group template first. If you delete only the resource group, the staging resource group (*/T\_*) that's used by VM Image Builder won't be cleaned up.

1. Remove the VM Image Builder template.

```
Remove-AzImageBuilderTemplate -ResourceGroupName $imageResourceGroup -Name vd10ImageTemplate
```

2. Delete the role assignment.

```
Remove-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $imageRoleDefName -  
Scope "/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"  
  
## Remove the definitions  
Remove-AzRoleDefinition -Name "$identityNamePrincipalId" -Force -Scope  
"/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"  
  
## Delete the identity  
Remove-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName -Force
```

3. Delete the resource group.

```
Remove-AzResourceGroup $imageResourceGroup -Force
```

## Next steps

To try more VM Image Builder examples, go to [GitHub](#).

# Azure Policy Regulatory Compliance controls for Azure VM Image Builder

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

[Regulatory Compliance in Azure Policy](#) provides initiative definitions, known as *built-ins*, for the compliance domains and security controls related to different compliance standards. Microsoft creates and manages these built-ins.

In this article, you can refer to a list of the compliance domains and security controls for Azure VM Image Builder. You can assign the built-ins for a security control individually, to help make your Azure resources compliant with the specific standard.

The title of each built-in policy definition links to the policy definition in the Azure portal. Use the link in the **Policy Version** column to view the source on the [Azure Policy GitHub repo](#).

## IMPORTANT

Each control is associated with one or more [Azure Policy](#) definitions. These policies might help you [assess compliance](#) with the control. However, there often isn't a one-to-one or complete match between a control and one or more policies. As such, **Compliant** in Azure Policy refers only to the policies themselves. This doesn't ensure that you're fully compliant with all requirements of a control. In addition, the compliance standard includes controls that aren't addressed by any Azure Policy definitions at this time. Therefore, compliance in Azure Policy is only a partial view of your overall compliance status. The associations between controls and Azure Policy Regulatory Compliance definitions for these compliance standards can change over time.

## Azure Security Benchmark

The [Azure Security Benchmark](#) provides recommendations on how you can secure your cloud solutions on Azure. To see how this service completely maps to the Azure Security Benchmark, see the [Azure Security Benchmark mapping files](#).

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Azure Security Benchmark](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Network Security	NS-2	Secure cloud services with network controls	<a href="#">VM Image Builder templates should use private link</a>	1.1.0

## FedRAMP High

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP High](#). For more information about this compliance standard, see [FedRAMP High](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-4	Information Flow Enforcement	VM Image Builder templates should use private link	1.1.0
Access Control	AC-17	Remote Access	VM Image Builder templates should use private link	1.1.0
Access Control	AC-17 (1)	Automated Monitoring / Control	VM Image Builder templates should use private link	1.1.0
System and Communications Protection	SC-7	Boundary Protection	VM Image Builder templates should use private link	1.1.0
System and Communications Protection	SC-7 (3)	Access Points	VM Image Builder templates should use private link	1.1.0

## FedRAMP Moderate

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP Moderate](#). For more information about this compliance standard, see [FedRAMP Moderate](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-4	Information Flow Enforcement	VM Image Builder templates should use private link	1.1.0
Access Control	AC-17	Remote Access	VM Image Builder templates should use private link	1.1.0
Access Control	AC-17 (1)	Automated Monitoring / Control	VM Image Builder templates should use private link	1.1.0
System and Communications Protection	SC-7	Boundary Protection	VM Image Builder templates should use private link	1.1.0
System and Communications Protection	SC-7 (3)	Access Points	VM Image Builder templates should use private link	1.1.0

## New Zealand ISM Restricted

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - New Zealand ISM Restricted](#). For more information about this compliance

standard, see [New Zealand ISM Restricted](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Infrastructure	INF-9	10.8.35 Security Architecture	<a href="#">VM Image Builder templates should use private link</a>	1.1.0

## NIST SP 800-53 Rev. 5

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NIST SP 800-53 Rev. 5](#). For more information about this compliance standard, see [NIST SP 800-53 Rev. 5](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-4	Information Flow Enforcement	<a href="#">VM Image Builder templates should use private link</a>	1.1.0
Access Control	AC-17	Remote Access	<a href="#">VM Image Builder templates should use private link</a>	1.1.0
Access Control	AC-17 (1)	Monitoring and Control	<a href="#">VM Image Builder templates should use private link</a>	1.1.0
System and Communications Protection	SC-7	Boundary Protection	<a href="#">VM Image Builder templates should use private link</a>	1.1.0
System and Communications Protection	SC-7 (3)	Access Points	<a href="#">VM Image Builder templates should use private link</a>	1.1.0

## NZ ISM Restricted v3.5

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NZ ISM Restricted v3.5](#). For more information about this compliance standard, see [NZ ISM Restricted v3.5](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Infrastructure	NZISM Security Benchmark INF-9	10.8.35 Security Architecture	<a href="#">VM Image Builder templates should use private link</a>	1.1.0

## Next steps

- Learn more about [Azure Policy Regulatory Compliance](#).
- See the built-ins on the [Azure Policy GitHub repo](#).

# Use Azure VM Image Builder for Linux VMs to access an existing Azure virtual network

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to use Azure VM Image Builder to create a basic, customized Linux image that has access to existing resources on a virtual network. The build virtual machine (VM) you create is deployed to a new or existing virtual network that you specify in your subscription. When you use an existing Azure virtual network, VM Image Builder doesn't require public network connectivity.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Set variables and permissions

For this task, you use some pieces of information repeatedly. Create some variables to store that information.

```
# set your environment variables here!!!!

# destination image resource group
imageResourceGroup=aibImageRG01

# location (see possible locations in main docs)
location=WestUS2

# your subscription
# get the current subID : 'az account show | grep id'
subscriptionID=$(az account show --query id --output tsv)

# name of the image to be created
imageName=aibCustomLinuxImg01

# image distribution metadata reference name
runOutputName=aibCustLinManImg01ro

# VNET properties (update to match your existing VNET, or leave as-is for demo)
# VNET name
vnetName=myexistingvnet01
# subnet name
subnetName=subnet01
# VNET resource group name
# NOTE! The VNET must always be in the same region as the Azure Image Builder service region.
vnetRgName=existingVnetRG
# Existing Subnet NSG Name or the demo will create it
nsgName=aibdemoNsg
```

Create the resource group.

```
az group create -n $imageResourceGroup -l $location
```

## Configure networking

If you don't have an existing virtual network, subnet, or network security group (NSG), use the following script to create one.

```

# Create a resource group

az group create -n $vnetRgName -l $location

# Create VNET

az network vnet create \
    --resource-group $vnetRgName \
    --name $vnetName --address-prefix 10.0.0.0/16 \
    --subnet-name $subnetName --subnet-prefix 10.0.0.0/24

# Create base NSG to simulate an existing NSG

az network nsg create -g $vnetRgName -n $nsgName

az network vnet subnet update \
    --resource-group $vnetRgName \
    --vnet-name $vnetName \
    --name $subnetName \
    --network-security-group $nsgName

# NOTE! The virtual network must always be in the same region as the Azure Image Builder service region.

```

## Add an NSG rule

This rule allows connectivity from the VM Image Builder load balancer to the proxy VM. Port 60001 is for Linux, and port 60000 is for Windows. The proxy VM connects to the build VM by using port 22 for Linux, or port 5986 for Windows.

```

az network nsg rule create \
    --resource-group $vnetRgName \
    --nsg-name $nsgName \
    -n AzureImageBuilderNsgRule \
    --priority 400 \
    --source-address-prefixes AzureLoadBalancer \
    --destination-address-prefixes VirtualNetwork \
    --destination-port-ranges 60000-60001 --direction inbound \
    --access Allow --protocol Tcp \
    --description "Allow Image Builder Private Link Access to Proxy VM"

```

## Disable private service policy on the subnet

Here's how:

```

az network vnet subnet update \
    --name $subnetName \
    --resource-group $vnetRgName \
    --vnet-name $vnetName \
    --disable-private-link-service-network-policies true

```

For more information, see [Azure VM Image Builder networking options](#).

## Modify the example template and create role

After you configure networking, you can modify the example template and create a role. Here's how:

```

# download the example and configure it with your vars

curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/1a_Creating_a_Custom_Linux_
Image_on_Existing_VNET/existingVNETLinux.json -o existingVNETLinux.json
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aib
RoleNetworking.json -o aibRoleNetworking.json
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aib
RoleImageCreation.json -o aibRoleImageCreation.json

sed -i -e "s/<subscriptionID>/$subscriptionID/g" existingVNETLinux.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" existingVNETLinux.json
sed -i -e "s/<region>/$location/g" existingVNETLinux.json
sed -i -e "s/<imageName>/$imageName/g" existingVNETLinux.json
sed -i -e "s/<runOutputName>/$runOutputName/g" existingVNETLinux.json

sed -i -e "s/<vnetName>/$vnetName/g" existingVNETLinux.json
sed -i -e "s/<subnetName>/$subnetName/g" existingVNETLinux.json
sed -i -e "s/<vnetRgName>/$vnetRgName/g" existingVNETLinux.json

sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" aibRoleImageCreation.json

sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleNetworking.json
sed -i -e "s/<vnetRgName>/$vnetRgName/g" aibRoleNetworking.json

```

## Set permissions on the resource group

VM Image Builder uses the [user identity](#) provided to inject the image into Azure Compute Gallery. In this example, you create an Azure role definition that can distribute the image to the gallery. The role definition is then assigned to the user identity.

```

# create user assigned identity for image builder
identityName=aibBuiUserId$(date +'%s')
az identity create -g $imageResourceGroup -n $identityName

# get identity id
imgBuilderCliId=$(az identity show -g $sigResourceGroup -n $identityName --query clientId -o tsv)

# get the user identity URI, needed for the template
imgBuilderId=/subscriptions/$subscriptionID/resourcegroups/$imageResourceGroup/providers/Microsoft.ManagedId
entity/userAssignedIdentities/$identityName

# update the template
sed -i -e "s%<imgBuilderId>%$imgBuilderId%g" existingVNETLinux.json

# make role name unique, to avoid clashes in the same Azure Active Directory domain
imageRoleDefName="Azure Image Builder Image Def"$(date +'%s')
netRoleDefName="Azure Image Builder Network Def"$(date +'%s')

# update the definitions
sed -i -e "s/Azure Image Builder Service Image Creation Role/$imageRoleDefName/g" aibRoleImageCreation.json
sed -i -e "s/Azure Image Builder Service Networking Role/$netRoleDefName/g" aibRoleNetworking.json

```

Instead of granting VM Image Builder lower granularity and increased privilege, you can create two roles. One role gives the builder permissions to create an image, and the other allows it to connect the build VM and load balancer to your virtual network.

```
# create role definitions
az role definition create --role-definition ./aibRoleImageCreation.json
az role definition create --role-definition ./aibRoleNetworking.json

# grant role definition to the user assigned identity
az role assignment create \
    --assignee $imgBuilderCliId \
    --role $imageRoleDefName \
    --scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup

az role assignment create \
    --assignee $imgBuilderCliId \
    --role $netRoleDefName \
    --scope /subscriptions/$subscriptionID/resourceGroups/$vnetRgName
```

For more information, see [Configure Azure VM Image Builder permissions by using the Azure CLI](#) or [Configure Azure VM Image Builder permissions by using PowerShell](#).

## Create the image

Submit the image configuration to VM Image Builder.

```
az resource create \
    --resource-group $imageResourceGroup \
    --properties @existingVNETLinux.json \
    --is-full-object \
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \
    -n existingVNETLinuxTemplate01

# Wait approximately 1-3 mins (validation, permissions etc.)
```

Start the image build.

```
az resource invoke-action \
    --resource-group $imageResourceGroup \
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \
    -n existingVNETLinuxTemplate01 \
    --action Run

# Wait approximately 15 mins
```

It can take a while to create the image and replicate it to both regions. Wait until this part is finished before moving on to creating a VM.

## Create a VM

Create a VM from the image version that was created by VM Image Builder.

```
az vm create \
    --resource-group $imageResourceGroup \
    --name aibImgVm0001 \
    --admin-username aibuser \
    --image $imageName \
    --location $location \
    --generate-ssh-keys
```

Use Secure Shell (SSH) to get into the VM.

```
ssh aibuser@<publicIpAddress>
```

You should see the image was customized with a *Message of the Day* as soon as your SSH connection is established!

```
*****
**      This VM was built from the:      **
**      !! AZURE VM IMAGE BUILDER Custom Image !!      **
**      You have just been Customized :-)      **
*****
```

## Clean up resources

If you want to recustomize the image version to create a new version of the same image, skip the next steps and go on to [Use Azure VM Image Builder to create another image version](#).

The following deletes the image that was created, along with all of the other resource files. Make sure you are finished with this deployment before deleting the resources.

When you delete gallery resources, you need to delete all of the image versions before you can delete the image definition used to create them. To delete a gallery, you first need to have deleted all of the image definitions in the gallery.

Delete the VM Image Builder template:

```
az resource delete \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n existingVNETLinuxTemplate01
```

Delete permissions assignments, roles, and identity:

```
az role assignment delete \
--assignee $imgBuilderCliId \
--role $imageRoleDefName \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup

az role assignment delete \
--assignee $imgBuilderCliId \
--role $netRoleDefName \
--scope /subscriptions/$subscriptionID/resourceGroups/$vnetRgName

az role definition delete --name "$imageRoleDefName"
az role definition delete --name "$netRoleDefName"

az identity delete --ids $imgBuilderId
```

Delete the resource group:

```
az group delete -n $imageResourceGroup
```

If you created a virtual network for this quickstart, you can delete the virtual network if it's no longer being used.

## Next steps



# Use Azure VM Image Builder to access an existing Azure virtual network

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article shows you how to use Azure VM Image Builder to create a basic, customized Windows image that has access to existing resources on a virtual network. The build virtual machine (VM) you create is deployed to a new or existing virtual network that you specify in your subscription. When you use an existing Azure virtual network, VM Image Builder doesn't require public network connectivity.

## Set variables and permissions

For this task, you use some pieces of information repeatedly. Create some variables to store that information.

```
# Step 1: Import module
Import-Module Az.Accounts

# Step 2: get existing context
$currentAzContext = Get-AzContext

# destination image resource group
$imageResourceGroup="aibImageRG"

# location (see possible locations in main docs)
$location="westus2"

## if you need to change your subscription: Get-AzSubscription / Select-AzSubscription -SubscriptionName

# get subscription, this will get your current subscription
$subscriptionID=$currentAzContext.Subscription.Id

# name of the image to be created
$imageName="win2019image01"

# image distribution metadata reference name
$runOutputName="win2019ManImg02ro"

# image template name
$imageTemplateName="window2019VnetTemplate03"

# distribution properties object name (runOutput), i.e. this gives you the properties of the managed image
# on completion
$runOutputName="winSvrSigR01"

# VNET properties (update to match your existing virtual network, or leave as-is for demo)
# VNET name
$vnetName="myexistingvnet01"
# subnet name
$subnetName="subnet01"
# VNET resource group name
$vnetRgName="existingVnetRG"
# Existing Subnet NSG Name or the demo will create it
$nsgName="aibdemoNsg"
# NOTE! The virtual network must always be in the same region as the VM Image Builder service region.
```

Create the resource group.

```
New-AzResourceGroup -Name $imageResourceGroup -Location $location
```

## Configure networking

If you don't have an existing virtual network, subnet, or network security group (NSG), use the following script to create one.

```
New-AzResourceGroup -Name $vnetRgName -Location $location

## Create base NSG to simulate an existing NSG
New-AzNetworkSecurityGroup -Name $nsgName -ResourceGroupName $vnetRgName -location $location

$nsg = Get-AzNetworkSecurityGroup -Name $nsgName -ResourceGroupName $vnetRgName

$subnet = New-AzVirtualNetworkSubnetConfig -Name $subnetName -AddressPrefix "10.0.1.0/24" -
PrivateLinkServiceNetworkPoliciesFlag "Disabled" -NetworkSecurityGroup $nsg

New-AzVirtualNetwork -Name $vnetName -ResourceGroupName $vnetRgName -Location $location -AddressPrefix
"10.0.0.0/16" -Subnet $subnet

## NOTE! The virtual network must always be in the same region as the VM Image Builder service region.
```

### Add an NSG rule

This rule allows connectivity from the VM Image Builder load balancer to the proxy VM. Port 60001 is for Linux, and port 60000 is for Windows. The proxy VM connects to the build VM by using port 22 for Linux, or port 5986 for Windows.

```
Get-AzNetworkSecurityGroup -Name $nsgName -ResourceGroupName $vnetRgName | Add-AzNetworkSecurityRuleConfig
-Name AzureImageBuilderAccess -Description "Allow Image Builder Private Link Access to Proxy VM" -Access
Allow -Protocol Tcp -Direction Inbound -Priority 400 -SourceAddressPrefix AzureLoadBalancer -SourcePortRange
* -DestinationAddressPrefix VirtualNetwork -DestinationPortRange 60000-60001 | Set-AzNetworkSecurityGroup
```

### Disable private service policy on the subnet

Here's how:

```
$virtualNetwork= Get-AzVirtualNetwork -Name $vnetName -ResourceGroupName $vnetRgName

($virtualNetwork | Select -ExpandProperty subnets | Where-Object {$_ .Name -eq $subnetName})
.privateLinkServiceNetworkPolicies = "Disabled"

$virtualNetwork | Set-AzVirtualNetwork
```

For more information, see [Azure VM Image Builder networking options](#).

## Modify the example template and create role

After you configure networking, you can modify the example template and create a role. Here's how:

```

$templateUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickquickstarts/1a_Creating_a
_Custom_Win_Image_on_Existing_VNET/existingVNETWindows.json"
$templateFilePath = "existingVNETWindows.json"

$aibRoleNetworkingUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating
_AIB_Security_Roles/aibRoleNetworking.json"
$aibRoleNetworkingPath = "aibRoleNetworking.json"

$aibRoleImageCreationUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creat
ing_AIB_Security_Roles/aibRoleImageCreation.json"
$aibRoleImageCreationPath = "aibRoleImageCreation.json"

# download configs
Invoke-WebRequest -Uri $templateUrl -OutFile $templateFilePath -UseBasicParsing

Invoke-WebRequest -Uri $aibRoleNetworkingUrl -OutFile $aibRoleNetworkingPath -UseBasicParsing

Invoke-WebRequest -Uri $aibRoleImageCreationUrl -OutFile $aibRoleImageCreationPath -UseBasicParsing

# update AIB image config template
((Get-Content -path $templateFilePath -Raw) -replace '<subscriptionID>', $subscriptionID) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<rgName>', $imageResourceGroup) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<region>', $location) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<runOutputName>', $runOutputName) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<imageName>', $imageName) | Set-Content -Path
$templateFilePath

((Get-Content -path $templateFilePath -Raw) -replace '<vnetName>', $vnetName) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<subnetName>', $subnetName) | Set-Content -Path
$templateFilePath
((Get-Content -path $templateFilePath -Raw) -replace '<vnetRgName>', $vnetRgName) | Set-Content -Path
$templateFilePath

```

## Create a user-assigned identity and set permissions

Next, you create a user-assigned identity and set permissions. Here's how:

```

# setup role def names, these need to be unique
$timeInt=$(get-date -UFormat "%s")
$imageRoleDefName="Azure Image Builder Image Def"+$timeInt
$networkRoleDefName="Azure Image Builder Network Def"+$timeInt
$idenityName="aibIdentity"+$timeInt

# create user identity
## Add AZ PS module to support AzUserAssignedIdentity
Install-Module -Name Az.ManagedServiceIdentity

# create identity
New-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $idenityName

$idenityNameResourceId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$idenityName).Id
$idenityNamePrincipalId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$idenityName).PrincipalId

# update template with identity
((Get-Content -path $templateFilePath -Raw) -replace '<imgBuilderId>',$idenityNameResourceId) | Set-Content
-Path $templateFilePath

# update the role definition names
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace 'Azure Image Builder Service Image Creation
Role',$imageRoleDefName) | Set-Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleNetworkingPath -Raw) -replace 'Azure Image Builder Service Networking
Role',$networkRoleDefName) | Set-Content -Path $aibRoleNetworkingPath

# update role definitions
((Get-Content -path $aibRoleNetworkingPath -Raw) -replace '<subscriptionID>',$subscriptionID) | Set-Content
-Path $aibRoleNetworkingPath
((Get-Content -path $aibRoleNetworkingPath -Raw) -replace '<vnetRgName>',$vnetRgName) | Set-Content -Path
$aibRoleNetworkingPath

((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<subscriptionID>',$subscriptionID) | Set-
Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<rgName>',$imageResourceGroup) | Set-Content
-Path $aibRoleImageCreationPath

# create role definitions from role configurations examples, this avoids granting contributor to the SPN
New-AzRoleDefinition -InputFile ./aibRoleImageCreation.json
New-AzRoleDefinition -InputFile ./aibRoleNetworking.json

# grant role definition to image builder user identity
New-AzRoleAssignment -ObjectId $idenityNamePrincipalId -RoleDefinitionName $imageRoleDefName -Scope
"/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"
New-AzRoleAssignment -ObjectId $idenityNamePrincipalId -RoleDefinitionName $networkRoleDefName -Scope
"/subscriptions/$subscriptionID/resourceGroups/$vnetRgName"

```

For more information, see [Configure Azure VM Image Builder permissions by using the Azure CLI](#) or [Configure Azure VM Image Builder permissions by using PowerShell](#).

## Create the image

Submit the image configuration to Azure VM Image Builder.

```

New-AzResourceGroupDeployment -ResourceGroupName $imageResourceGroup -TemplateFile $templateFilePath -api-
version "2020-02-14" -imageTemplateName $imageTemplateName -svclocation $location

# note this will take minute, as validation is run (security / dependencies etc.)

```

Start the image build.

```
Invoke-AzResourceAction -ResourceName $imageTemplateName -ResourceGroupName $imageResourceGroup -  
 ResourceType Microsoft.VirtualMachineImages/imageTemplates -ApiVersion "2020-02-14" -Action Run -Force
```

## Get build status and properties

First, you query the image template for current or last run status, and for image template settings.

```
$managementEp = $currentAzureContext.Environment.ResourceManagerUrl  
  
$urlBuildStatus = [System.String]::Format("{0}subscriptions/{1}/resourceGroups/$imageResourceGroup/providers/Microsoft.VirtualMachineImages/imageTemplates/{2}?api-version=2020-02-14", $managementEp, $currentAzureContext.Subscription.Id,$imageTemplateName)  
  
$buildStatusResult = Invoke-WebRequest -Method GET -Uri $urlBuildStatus -UseBasicParsing -Headers  
@{ "Authorization"= ("Bearer " + $accessToken)} -ContentType application/json  
$buildJsonStatus =$buildStatusResult.Content  
$buildJsonStatus
```

The image build for this example takes approximately 50 minutes (including multiple reboots and Windows updates). When you query the status, look for `lastRunStatus`. The following code shows that the build is still running. If it had completed successfully, it would show `succeeded`.

```
"lastRunStatus": {  
    "startTime": "2019-08-21T00:39:40.61322415Z",  
    "endTime": "0001-01-01T00:00:00Z",  
    "runState": "Running",  
    "runSubState": "Building",  
    "message": ""  
},
```

## Query the distribution properties

If you're distributing to a VHD location, need managed image location properties, or Azure Compute Gallery replications status, you need to query `runOutput`. Every time you have a distribution target, you will have a unique `runOutput`, to describe properties of the distribution type.

```
$managementEp = $currentAzureContext.Environment.ResourceManagerUrl  
$urlRunOutputStatus = [System.String]::Format("{0}subscriptions/{1}/resourceGroups/$imageResourceGroup/providers/Microsoft.VirtualMachineImages/imageTemplates/$imageTemplateName/runOutputs/{2}?api-version=2020-02-14", $managementEp,  
$currentAzureContext.Subscription.Id, $runOutputName)  
  
$runOutStatusResult = Invoke-WebRequest -Method GET -Uri $urlRunOutputStatus -UseBasicParsing -Headers  
@{ "Authorization"= ("Bearer " + $accessToken)} -ContentType application/json  
$runOutJsonStatus =$runOutStatusResult.Content  
$runOutJsonStatus
```

## Create a VM

Now that the build is finished, you can build a VM from the image. Use the examples from the [PowerShell New-AzVM documentation](#).

## Clean up tasks

You can now delete the image template artifact, the role assignment, and the resource groups if you want to.

Here's how to delete the image template artifact:

```
# Get ResourceID of the Image Template
$resTemplateId = Get-AzResource -ResourceName $imageTemplateName -ResourceGroupName $imageResourceGroup -
 ResourceType Microsoft.VirtualMachineImages/imageTemplates -ApiVersion "2020-02-14"

### Delete Image Template Artifact
Remove-AzResource -ResourceId $resTemplateId.ResourceId -Force
```

Here's how to delete the role assignment:

```
## remove role assignments
Remove-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $imageRoleDefName -Scope
"/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"
Remove-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $networkRoleDefName -Scope
"/subscriptions/$subscriptionID/resourceGroups/$vnetRgName"

## remove definitions
Remove-AzRoleDefinition -Id $imageRoleDefObjId -Force
Remove-AzRoleDefinition -Id $networkRoleObjId -Force

## delete identity
Remove-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName -Force
```

Here's how to delete resource groups:

```
Remove-AzResourceGroup $imageResourceGroup -Force

# delete VNET created
# BEWARE! In this example, you have either used an existing virtual network or created one for this example.
# Do not delete your existing virtual network. If you want to delete the virtual network resource group used
# in this example '$vnetRgName', modify the preceding code.
```

## Next steps

[Azure Compute Galleries](#)

# Azure VM Image Builder networking options

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

With Azure VM Image Builder, you choose to deploy the service with or without an existing virtual network. The following sections provide more details about this choice.

## Deploy without specifying an existing virtual network

If you don't specify an existing virtual network, VM Image Builder creates one, along with a subnet, in the staging resource group. The service uses a public IP resource with a network security group to restrict inbound traffic. The public IP facilitates the channel for commands during the image build. After the build completes, the virtual machine (VM), public IP, disks, and virtual network are deleted. To use this option, don't specify any virtual network properties.

## Deploy using an existing VNET

If you specify a virtual network and subnet, VM Image Builder deploys the build VM to your chosen virtual network. You can access resources that are accessible on your virtual network. You can also create a siloed virtual network, unconnected to any other virtual network. If you specify a virtual network, VM Image Builder doesn't use a public IP address. Communication from VM Image Builder to the build virtual machine uses Azure Private Link.

For more information, see one of the following examples:

- [Use Azure VM Image Builder for Windows VMs allowing access to an existing Azure virtual network](#)
- [Use Azure VM Image Builder for Linux VMs allowing access to an existing Azure virtual network](#)

### What is Azure Private Link?

Azure Private Link provides private connectivity from a virtual network to Azure platform as a service (PaaS), or to customer-owned or Microsoft partner services. It simplifies the network architecture, and secures the connection between endpoints in Azure by eliminating data exposure to the public internet. For more information, see the [Private Link documentation](#).

### Required permissions for an existing virtual network

VM Image Builder requires specific permissions to use an existing virtual network. For more information, see [Configure Azure VM Image Builder permissions by using the Azure CLI](#) or [Configure Azure VM Image Builder permissions by using PowerShell](#).

### What is deployed during an image build?

If you use an existing virtual network, VM Image Builder deploys an additional VM (a *proxy* VM), and a load balancer (Azure Load Balancer). These are connected to Private Link. Traffic from the VM Image Builder service goes across the private link to the load balancer. The load balancer communicates to the proxy VM by using port 60001 for Linux, or port 60000 for Windows. The proxy forwards commands to the build VM by using port 22 for Linux, or port 5986 for Windows.

#### NOTE

The virtual network must be in the same region as the VM Image Builder service region.

## Why deploy a proxy VM?

When a VM without a public IP is behind an internal load balancer, it doesn't have internet access. The load balancer used for the virtual network is internal. The proxy VM allows internet access for the build VM during builds. You can use the associated network security groups to restrict the build VM access.

The deployed proxy VM size is *Standard A1\_v2*, in addition to the build VM. The VM Image Builder service uses the proxy VM to send commands between the service and the build VM. You can't change the proxy VM properties (this restriction includes the size and the operating system).

## Image template parameters to support the virtual network

```
"VirtualNetworkConfig": {  
    "name": "",  
    "subnetName": "",  
    "resourceGroupName": ""  
},
```

SETTING	DESCRIPTION
<code>name</code>	(Optional) The name of a pre-existing virtual network.
<code>subnetName</code>	The name of the subnet within the specified virtual network. You must specify this setting if, and only if, the <code>name</code> setting is specified.
<code>resourceGroupName</code>	The name of the resource group containing the specified virtual network. You must specify this setting if, and only if, the <code>name</code> setting is specified.

Private Link requires an IP from the specified virtual network and subnet. Currently, Azure doesn't support network policies on these IPs. Hence, you must disable network policies on the subnet. For more information, see the [Private Link documentation](#).

## Checklist for using your virtual network

- Allow Azure Load Balancer to communicate with the proxy VM in a network security group.
  - [Azure CLI example](#)
  - [PowerShell example](#)
- Disable the private service policy on the subnet.
  - [Azure CLI example](#)
  - [PowerShell example](#)
- Allow VM Image Builder to create a load balancer, and add VMs to the virtual network.
  - [Azure CLI example](#)
  - [PowerShell example](#)
- Allow VM Image Builder to read and write source images, and create images.
  - [Azure CLI example](#)
  - [PowerShell example](#)
- Ensure that you're using a virtual network in the same region as the VM Image Builder service region.

## Next steps

[Azure VM Image Builder overview](#)

# Configure Azure VM Image Builder permissions by using the Azure CLI

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you register for Azure VM Image Builder, this grants the service permission to create, manage, and delete a staging resource group. The service also has rights to add resources to a resource group, required for the image build. During a successful registration, your subscription gets access to a VM Image Builder service principal name (SPN).

If you want VM Image Builder to distribute images, you need to create a user-assigned identity in Azure, with permissions to read and write images. For example, you might want to distribute images to managed images or to Azure Compute Gallery. If you're accessing Azure storage, then the user-assigned identity you create needs permissions to read private or public containers.

You must set up permissions and privileges prior to building an image. The following sections detail how to configure possible scenarios by using the Azure CLI.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Create a user-assigned managed identity

VM Image Builder requires you to create an [Azure user-assigned managed identity](#). VM Image Builder uses this identity to read images, write images, and access Azure storage accounts. You grant the identity permission to do specific actions in your subscription.

### NOTE

User-assigned managed identity is the correct way to grant permissions to the image resource groups. The SPN is deprecated for this purpose.

The following example shows you how to create an Azure user-assigned managed identity. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Resource group>	The resource group where you want to create the user-assigned managed identity.

```
identityName="aibIdentity"
imageResourceGroup=<Resource group>

az identity create \
--resource-group $imageResourceGroup \
--name $identityName
```

For more information, see [Azure user-assigned managed identity](#).

## Allow VM Image Builder to distribute images

For VM Image Builder to distribute images, the service must be allowed to inject the images into resource groups. To grant the required permissions, create a user-assigned managed identity, and grant it rights on the resource group where the image is built. VM Image Builder doesn't have permission to access resources in other resource groups in the subscription. You need to take explicit actions to allow access, to prevent your builds from failing.

You don't need to grant the user-assigned managed identity contributor rights on the resource group to distribute images. However, the user-assigned managed identity needs the following Azure [Actions](#) permissions in the distribution resource group:

```
Microsoft.Compute/images/write
Microsoft.Compute/images/read
Microsoft.Compute/images/delete
```

If you want to distribute to Azure Compute Gallery, you also need:

```
Microsoft.Compute/galleries/read
Microsoft.Compute/galleries/images/read
Microsoft.Compute/galleries/images/versions/read
Microsoft.Compute/galleries/images/versions/write
```

## Permission to customize existing images

For VM Image Builder to build images from source custom images, the service must be allowed to read the images into these resource groups. To grant the required permissions, create a user-assigned managed identity, and grant it rights on the resource group where the image is located.

Here's how you build from an existing custom image:

```
Microsoft.Compute/galleries/read
```

Here's how you build from an existing Azure Compute Gallery version:

```
Microsoft.Compute/galleries/read  
Microsoft.Compute/galleries/images/read  
Microsoft.Compute/galleries/images/versions/read
```

## Permission to customize images on your virtual networks

VM Image Builder has the capability to deploy and use an existing virtual network in your subscription, thus allowing customizations access to connected resources.

You don't need to grant the user-assigned managed identity contributor rights on the resource group to deploy a VM to an existing virtual network. However, the user-assigned managed identity needs the following Azure Actions permissions on the virtual network resource group:

```
Microsoft.Network/virtualNetworks/read  
Microsoft.Network/virtualNetworks/subnets/join/action
```

## Create an Azure role definition

The following examples create an Azure role definition from the actions described in the previous sections. The examples are applied at the resource group level. Evaluate and test if the examples are granular enough for your requirements.

The image actions allow read and write. Decide what is appropriate for your environment. For example, create a role to allow VM Image Builder to read images from resource group *example-rg-1*, and write images to resource group *example-rg-2*.

### Custom image Azure role example

The following example creates an Azure role to use and distribute a source custom image. You then grant the custom role to the user-assigned managed identity for VM Image Builder.

To simplify the replacement of values in the example, set the following variables first. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Subscription ID>	Your Azure subscription ID.
<Resource group>	Resource group for the custom image.

```

# Subscription ID - You can get this using `az account show | grep id` or from the Azure portal.
subscriptionID=$(az account show --query id --output tsv)
# Resource group - image builder will only support creating custom images in the same Resource Group as the
# source managed image.
imageResourceGroup=<Resource group>
identityName="aibIdentity"

# Use *cURL* to download the a sample JSON description
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aib
RoleImageCreation.json -o aibRoleImageCreation.json

# Create a unique role name to avoid clashes in the same Azure Active Directory domain
imageRoleDefName="Azure Image Builder Image Def$(date +'%s')

# Update the JSON definition using stream editor
sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" aibRoleImageCreation.json
sed -i -e "s/Azure Image Builder Service Image Creation Role/$imageRoleDefName/g" aibRoleImageCreation.json

# Create a custom role from the sample aibRoleImageCreation.json description file.
az role definition create --role-definition ./aibRoleImageCreation.json

# Get the user-assigned managed identity id
imgBuilderCliId=$(az identity show -g $imageResourceGroup -n $identityName --query clientId -o tsv)

# Grant the custom role to the user-assigned managed identity for Azure Image Builder.
az role assignment create \
--assignee $imgBuilderCliId \
--role $imageRoleDefName \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup

```

### Existing virtual network Azure role example

The following example creates an Azure role to use and distribute an existing virtual network image. You then grant the custom role to the user-assigned managed identity for VM Image Builder.

To simplify the replacement of values in the example, set the following variables first. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Subscription ID>	Your Azure subscription ID.
<Resource group>	The virtual network resource group

```

# Subscription ID - You can get this using `az account show | grep id` or from the Azure portal.
subscriptionID=$(az account show --query id --output tsv)
VnetResourceGroup=<Resource group>
identityName="aibIdentity"

# Use *cURL* to download the a sample JSON description
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aib
RoleNetworking.json -o aibRoleNetworking.json

# Create a unique role name to avoid clashes in the same domain
netRoleDefName="Azure Image Builder Network Def$(date +'%s')

# Update the JSON definition using stream editor
sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleNetworking.json
sed -i -e "s/<vnetRgName>/$vnetRgName/g" aibRoleNetworking.json
sed -i -e "s/Azure Image Builder Service Networking Role/$netRoleDefName/g" aibRoleNetworking.json

# Create a custom role from the aibRoleNetworking.json description file.
az role definition create --role-definition ./aibRoleNetworking.json

# Get the user-assigned managed identity id
imgBuilderCliId=$(az identity show -g $imageResourceGroup -n $identityName --query clientId -o tsv)

# Grant the custom role to the user-assigned managed identity for Azure Image Builder.
az role assignment create \
--assignee $imgBuilderCliId \
--role $netRoleDefName \
--scope /subscriptions/$subscriptionID/resourceGroups/$VnetResourceGroup

```

## Using managed identity for Azure Storage access

If you want to authenticate with Azure Storage and use private containers, VM Image Builder needs a user-assigned managed identity. VM Image Builder uses the identity to authenticate with Azure Storage.

### NOTE

VM Image Builder only uses the identity at the time that you submit the image template. The build VM doesn't have access to the identity during image build.

Use the Azure CLI to create the user-assigned managed identity:

```

az role assignment create \
--assignee <Image Builder client ID> \
--role "Storage Blob Data Reader" \
--scope /subscriptions/<Subscription ID>/resourceGroups/<Resource
group>/providers/Microsoft.Storage/storageAccounts/$scriptStorageAcc/blobServices/default/containers/<Storag
e account container>

```

In the VM Image Builder template, provide the user-assigned managed identity:

```
"type": "Microsoft.VirtualMachineImages/imageTemplates",
"apiVersion": "2020-02-14",
"location": "<Region>",
..
"identity": {
    "type": "UserAssigned",
    "userAssignedIdentities": {
        "<Image Builder ID>": {}
    }
}
```

Replace the following placeholder settings:

SETTING	DESCRIPTION
<Region>	Template region
<Resource group>	Resource group
<Storage account container>	Storage account container name
<Subscription ID>	Azure subscription

For more information, see [Create an image and use a user-assigned managed identity to access files in Azure Storage](#). You learn how to create and configure the user-assigned managed identity to access a storage account.

## Next steps

[Azure VM Image Builder overview](#)

# Configure Azure VM Image Builder permissions by using PowerShell

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you register for Azure VM Image Builder, this grants the service permission to create, manage, and delete a staging resource group. The service also has rights to add resources to a resource group, required for the image build. During a successful registration, your subscription gets access to a VM Image Builder service principal name (SPN).

If you want VM Image Builder to distribute images, you need to create a user-assigned identity in Azure, with permissions to read and write images. For example, you might want to distribute images to managed images or to Azure Compute Gallery. If you're accessing Azure Storage, then the user-assigned identity you create needs permissions to read private or public containers.

You must set up permissions and privileges prior to building an image. The following sections detail how to configure possible scenarios by using PowerShell.

## Create a user-assigned managed identity

VM Image Builder requires you to create an [Azure user-assigned managed identity](#). VM Image Builder uses this identity to read images, write images, and access Azure Storage accounts. You grant the identity permission to do specific actions in your subscription.

### NOTE

User-assigned managed identity is the correct way to grant permissions to the image resource groups. The SPN is deprecated for this purpose.

The following example shows you how to create an Azure user-assigned managed identity. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Resource group>	The resource group where you want to create the user-assigned managed identity.

```
## Add AZ PS module to support AzUserAssignedIdentity
Install-Module -Name Az.ManagedServiceIdentity

$parameters = @{
    Name = 'aibIdentity'
    ResourceGroupName = '<Resource group>'
}
# create identity
New-AzUserAssignedIdentity @parameters
```

For more information, see [Azure user-assigned managed identity](#).

## Allow VM Image Builder to distribute images

For VM Image Builder to distribute images, the service must be allowed to inject the images into resource groups. To grant the required permissions, create a user-assigned managed identity, and grant it rights on the resource group where the image is built. VM Image Builder doesn't have permission to access resources in other resource groups in the subscription. You need to take explicit actions to allow access, to prevent your builds from failing.

You don't need to grant the user-assigned managed identity contributor rights on the resource group to distribute images. However, the user-assigned managed identity needs the following Azure **Actions** permissions in the distribution resource group:

```
Microsoft.Compute/images/write  
Microsoft.Compute/images/read  
Microsoft.Compute/images/delete
```

If you want to distribute to Azure Compute Gallery, you also need:

```
Microsoft.Compute/galleries/read  
Microsoft.Compute/galleries/images/read  
Microsoft.Compute/galleries/images/versions/read  
Microsoft.Compute/galleries/images/versions/write
```

## Permission to customize existing images

For VM Image Builder to build images from source custom images, the service must be allowed to read the images into these resource groups. To grant the required permissions, create a user-assigned managed identity, and grant it rights on the resource group where the image is located.

Here's how you build from an existing custom image:

```
Microsoft.Compute/galleries/read
```

Here's how you build from an existing Azure Compute Gallery version:

```
Microsoft.Compute/galleries/read  
Microsoft.Compute/galleries/images/read  
Microsoft.Compute/galleries/images/versions/read
```

## Permission to customize images on your virtual networks

VM Image Builder has the capability to deploy and use an existing virtual network in your subscription, thus allowing customizations access to connected resources.

You don't need to grant the user-assigned managed identity contributor rights on the resource group to deploy a VM to an existing virtual network. However, the user-assigned managed identity needs the following Azure **Actions** permissions on the virtual network resource group:

```
Microsoft.Network/virtualNetworks/read  
Microsoft.Network/virtualNetworks/subnets/join/action
```

## Create an Azure role definition

The following examples create an Azure role definition from the actions described in the previous sections. The examples are applied at the resource group level. Evaluate and test if the examples are granular enough for your requirements.

The image actions allow read and write. Decide what is appropriate for your environment. For example, create a role to allow VM Image Builder to read images from resource group *example-rg-1*, and write images to resource group *example-rg-2*.

### Custom image Azure role example

The following example creates an Azure role to use and distribute a source custom image. You then grant the custom role to the user-assigned managed identity for VM Image Builder.

To simplify the replacement of values in the example, set the following variables first. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Subscription ID>	Your Azure subscription ID.
<Resource group>	Resource group for the custom image.

```

$sub_id = "<Subscription ID>"
# Resource group - image builder will only support creating custom images in the same Resource Group as the
source managed image.
$imageResourceGroup = "<Resource group>"
$identityName = "aibIdentity"

# Use a web request to download the sample JSON description
$sample_uri="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleImageCreation.json"
$role_definition="aibRoleImageCreation.json"

Invoke-WebRequest -Uri $sample_uri -Outfile $role_definition -UseBasicParsing

# Create a unique role name to avoid clashes in the same Azure Active Directory domain
$timeInt=$(get-date -UFormat "%s")
$imageRoleDefName="Azure Image Builder Image Def"+$timeInt

# Update the JSON definition placeholders with variable values
((Get-Content -path $role_definition -Raw) -replace '<subscriptionID>', $sub_id) | Set-Content -Path
$role_definition
((Get-Content -path $role_definition -Raw) -replace '<rgName>', $imageResourceGroup) | Set-Content -Path
$role_definition
((Get-Content -path $role_definition -Raw) -replace 'Azure Image Builder Service Image Creation Role',
$imageRoleDefName) | Set-Content -Path $role_definition

# Create a custom role from the aibRoleImageCreation.json description file.
New-AzRoleDefinition -InputFile $role_definition

# Get the user-identity properties
$identityNameResourceId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).Id
$identityNamePrincipalId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).PrincipalId

# Grant the custom role to the user-assigned managed identity for Azure Image Builder.
$parameters = @{
    ObjectId = $identityNamePrincipalId
    RoleDefinitionName = $imageRoleDefName
    Scope = '/subscriptions/' + $sub_id + '/resourceGroups/' + $imageResourceGroup
}
New-AzRoleAssignment @parameters

```

## Existing virtual network Azure role example

The following example creates an Azure role to use and distribute an existing virtual network image. You then grant the custom role to the user-assigned managed identity for VM Image Builder.

To simplify the replacement of values in the example, set the following variables first. Replace the placeholder settings to set your variables.

SETTING	DESCRIPTION
<Subscription ID>	Your Azure subscription ID.
<Resource group>	The virtual network resource group.

```

$sub_id = "<Subscription ID>"
$res_group = "<Resource group>"
$identityName = "aibIdentity"

# Use a web request to download the sample JSON description
$sample_uri="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleNetworking.json"
$role_definition="aibRoleNetworking.json"

Invoke-WebRequest -Uri $sample_uri -Outfile $role_definition -UseBasicParsing

# Create a unique role name to avoid clashes in the same AAD domain
$timeInt=$(get-date -UFormat "%s")
$networkRoleDefName="Azure Image Builder Network Def"+$timeInt

# Update the JSON definition placeholders with variable values
((Get-Content -path $role_definition -Raw) -replace '<subscriptionID>', $sub_id) | Set-Content -Path $role_definition
((Get-Content -path $role_definition -Raw) -replace '<vnetRgName>', $res_group) | Set-Content -Path $role_definition
((Get-Content -path $role_definition -Raw) -replace 'Azure Image Builder Service Networking Role', $networkRoleDefName) | Set-Content -Path $role_definition

# Create a custom role from the aibRoleNetworking.json description file
New-AzRoleDefinition -InputFile $role_definition

# Get the user-identity properties
$identityNameResourceId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName).Id
$identityNamePrincipalId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName).PrincipalId

# Assign the custom role to the user-assigned managed identity for Azure Image Builder
$parameters = @{
    ObjectId = $identityNamePrincipalId
    RoleDefinitionName = $networkRoleDefName
    Scope = '/subscriptions/' + $sub_id + '/resourceGroups/' + $res_group
}

New-AzRoleAssignment @parameters

```

## Next steps

[Azure VM Image Builder overview](#)

# Azure VM Image Builder service DevOps task (preview)

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In this article, you learn how to use an Azure DevOps task to inject build artifacts into a virtual machine (VM) image, so that you can install and configure your application and operating system.

## DevOps task versions

At this time, there are two Azure VM Image Builder DevOps tasks:

- *Stable* VM Image Builder task: The latest stable build that's been tested, and reports no [General Data Protection Regulation \(GDPR\)](#) issues.
- *Unstable* VM Image Builder task: We offer a so-called *unstable* task so that you can test the latest updates and features before we release the task code as *stable*. After about a week, if there are no customer-reported or telemetry issues, we promote the task code to *stable*.

## Prerequisites

### NOTE

The VM Image Builder task doesn't currently support Windows Restart or running elevated commands as Administrator. That is, the task isn't suitable for Azure Virtual Desktop scenarios or Windows customizations that require those features. To use DevOps with VM Image Builder, nest the template within an Azure Resource Manager task, and use Azure CLI or PowerShell tasks.

Before you begin, you must:

- Install [Stable DevOps task from Visual Studio Marketplace](#).
- Have an Azure DevOps Services (formerly Visual Studio Team Services, or VSTS) account, and a Build Pipeline created.
- Register and enable the VM Image Builder feature requirements in the subscription that's used by the pipelines:
  - [Azure PowerShell](#)
  - [The Azure CLI](#)
- Create a standard Azure storage account in the source image resource group. You can use other resource groups or storage accounts. The storage account is used transfer the build artifacts from the DevOps task to the image.

```
# Azure PowerShell
$timeInt=$(get-date -UFormat "%s")
$storageAccName="aibstorage"+$timeInt
$location=westus
# Create a storage account and blob in the resource group
New-AzStorageAccount -ResourceGroupName $strResourceGroup -Name $storageAccName -Location $location -
SkuName Standard_LRS
```

```
# The Azure CLI
location=westus
scriptStorageAcc=aibstordot$(date +'%s')
# Create a storage account and blob in the resource group
az storage account create -n $scriptStorageAcc -g $strResourceGroup -l $location --sku Standard_LRS
```

## Add a task to the release pipeline

1. Select **Release Pipeline > Edit**.
2. On the User Agent, select the plus sign (+) to add and search for **Image Builder**.
3. Select **Add**.

In the following sections, set the task properties.

### Azure subscription

In the dropdown list, select the subscription that you want VM Image Builder to run. Use the subscription where your source images are stored and the images are to be distributed. You need to grant the VM Image Builder contributor access to the subscription or resource group.

### Resource group

Use the resource group where the temporary image template artifact will be stored. When you create a template artifact, another temporary VM Image Builder resource group,

`IT_<DestinationResourceGroup>_<TemplateName>_guid`, is created. The temporary resource group stores the image metadata, such as scripts. At the end of the task, the image template artifact and temporary VM Image Builder resource group is deleted.

### Location

The location is the region where VM Image Builder will run. Only a set number of [regions](#) are supported. The source images must be present in this location. For example, if you're using Azure Compute Gallery (formerly Shared Image Gallery), a replica must exist in that region.

### Managed identity (required)

VM Image Builder requires a managed identity, which it uses to read source custom images, connect to Azure Storage, and create custom images. For more information, see [Learn about VM Image Builder](#).

### Virtual network support

You can configure the created VM to be in a specific virtual network. When you configure the task, provide the resource ID of a pre-existing subnet in the **VNet Configuration (Optional)** input field. Omit the resource ID if no specific virtual network needs to be used. For more information, see [Azure VM Image Builder service networking options](#).

### Source

The source images must be of the supported VM Image Builder operating systems. You can choose existing custom images in the same region that VM Image Builder is running from:

- Managed Image: Pass in the resource ID. For example:

```
/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/images/<imageName>
```

- Compute Gallery: Pass in the resource ID of the image version. For example:

```
/subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup/providers/Microsoft.Compute/galleries/$sigName/images/$imageDefName/versions/<versionNumber>
```

If you need to get the latest Compute Gallery version, use an Azure PowerShell or Azure CLI task to get it and set a DevOps variable. Use the variable in the VM Image Builder DevOps task. For more information, see the examples in [Get the latest image version resource ID](#).

- (Marketplace) Base image: Use the dropdown list of popular images, which always uses the latest version of the supported operating systems.

If the base image isn't in the list, you can specify the exact image by using `Publisher:Offer:Sku`.

(Optional) Base image version: You can supply the version of the image that you want to use. The default version is `latest`.

## Customize

The following sections discuss various ways to customize tasks.

### Provisioner

Initially, two customizers are supported, Shell and PowerShell. Only inline is supported. If you want to download scripts, you can pass inline commands to do so.

For your operating system, select PowerShell or Shell.

### The Windows Update task

For Windows only, the task runs Windows Update at the end of the customizations. It also handles the required reboots.

The task runs the following Windows Update configuration:

```
"type": "WindowsUpdate",
"searchCriteria": "IsInstalled=0",
"filters": [
    "exclude:$_.Title -like '*Preview*'",
    "include:$true"
```

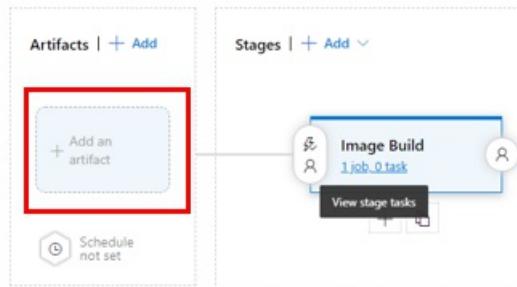
The task installs important and recommended Windows Updates that aren't *preview* versions.

### Handling reboots

The DevOps task doesn't currently support rebooting Windows builds. If you try to reboot with PowerShell code, the build fails. However, you can use code to reboot Linux builds.

### Build path

The task is designed to be able to inject DevOps Build release artifacts into the image. To make this work, you need to set up a build pipeline. In the release pipeline setup, add the repo of the build artifacts.



Select the **Build Path** button to choose the build folder that you want to be placed on the image. The VM Image Builder task copies all the files and directories within it. When the image is being created, VM Image Builder deploys the files and directories into different paths, depending on the operating system.

#### IMPORTANT

When you're adding a repo artifact, you might find that the directory name is prefixed with an underscore character (\_). The underscore can cause issues with the inline commands. Be sure to use the appropriate quotation marks in the commands.

The following example explains how this works:

Select a file or folder X

```

        ▲ └ Linked artifacts
          └ _ImageBuilding (Azure Repos Git)
            └ webapp
              └ webconfig.ps1
              └ azure-pipelines.yml
              └ README.md
    
```

- For Windows: Files exist in the *C*: drive. A directory named *buildArtifacts* is created, which includes the *webapp* directory.
- For Linux: Files exist in the */tmp* directory. The *webapp* directory is created, which includes all the files and directories. Because this is a temporary directory, you must move the files out of it. Otherwise, they'll be deleted.

#### Inline customization script

- For Windows: You can enter PowerShell inline commands, separated by commas. If you want to run a script in your build directory, you can use:

```
& 'c:\buildArtifacts\webapp\webconfig.ps1'
```

You can reference multiple scripts or add more commands. For example:

```
& 'c:\buildArtifacts\webapp\webconfig.ps1'
& 'c:\buildArtifacts\webapp\installAgent.ps1'
```

- For Linux: The build artifacts are put into the */tmp* directory. However, on many Linux operating systems, on a reboot, the */tmp* directory contents are deleted. If you want the artifacts to exist in the image, you must create another directory and copy them over. For example:

```
sudo mkdir /lib/buildArtifacts  
sudo cp -r "/tmp/_ImageBuilding/webapp" /lib/buildArtifacts/.
```

If you're OK with using the `/tmp` directory, you can run the script by using the following code:

```
# Grant execute permissions to run scripts  
sudo chmod +x "/tmp/_ImageBuilding/webapp/coreConfig.sh"  
echo "running script"  
sudo . "/tmp/AppsAndImageBuilderLinux/_WebApp/coreConfig.sh"
```

#### What happens to the build artifacts after the image build?

##### NOTE

VM Image Builder doesn't automatically remove the build artifacts. We strongly suggest that you always use code to remove the build artifacts.

- For Windows: VM Image Builder deploys files to the `C:\buildArtifacts` directory. Because the directory is persisted, you must remove it by running a script. For example:

```
# Clean up buildArtifacts directory  
Remove-Item -Path "C:\buildArtifacts\*" -Force -Recurse  
  
# Delete the buildArtifacts directory  
Remove-Item -Path "C:\buildArtifacts" -Force
```

- For Linux: The build artifacts are put into the `/tmp` directory. However, on many Linux operating systems, the `/tmp` directory contents are deleted on reboot. We suggest that you use code to remove the contents and not rely on the operating system to remove the contents. For example:

```
sudo rm -R "/tmp/AppsAndImageBuilderLinux"
```

#### Total length of image build

Total length can't be changed in the DevOps pipeline task yet. It uses the default of 240 minutes. If you want to increase the `buildTimeoutInMinutes`, you can use an Azure CLI task in the release pipeline. Configure the task to copy a template and submit it. For an example solution, see [Use environment variables and parameters with VM Image Builder](#), or use Azure PowerShell.

#### Storage account

Select the storage account you created in the prerequisites. If you don't see it in the list, VM Image Builder doesn't have permissions to it.

When the build starts, VM Image Builder creates a container called `imagebuilder-vststask`, where the build artifacts from the repo are stored.

##### NOTE

You need to manually delete the storage account or container after each build.

#### Distribute

The following three distribute types are supported.

##### Managed image

- Resource ID:

```
/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/images/<imageName>
```

- Locations

#### Azure Compute Gallery

The Compute Gallery must already exist.

- Resource ID:

```
/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/<gallerName>/images/<imageDefName>
```

- Regions: A list of regions, comma separated. For example, `westus`, `eastus`, `centralus`.

#### Virtual hard disk

You can't pass any values to this. VM Image Builder emits the virtual hard disk VHD to the temporary VM Image Builder resource group, `IT_{DestinationResourceGroup}_{TemplateName}`, in the `vhds` container. When you start the release build, VM Image Builder emits logs. When VM Image Builder has finished, it emits the VHD URL.

#### Optional settings

You can override the [VM size](#) setting from its default size of `Standard_D1_v2`. You might want to do so to reduce total customization time. Or you might want to create images that depend on certain VM sizes, such as GPU (graphics processing unit), HPC (high-performance computing), and so on.

## How the task works

When you create the release, the task creates a container in the storage account, named `imagebuilder-vststask`. It zips (compresses) and uploads your build artifacts and creates a shared access signature token for the zip file.

The task uses the properties that are passed to the task to create the VM Image Builder template artifact. The task does the following:

- Downloads the build artifact zip file and any other associated scripts. The files are saved in a storage account in the temporary VM Image Builder resource group `IT_{DestinationResourceGroup}_{TemplateName}`.
- Creates a template that's prefixed with `t_` and a 10-digit monotonic integer. The template is saved to the resource group that you selected, and it exists for the duration of the build in the resource group.

Example output:

```
start reading task parameters...
found build at: /home/vsts/work/r1/a/_ImageBuilding/webapp
end reading parameters
getting storage account details for aibstordot1556933914
created archive /home/vsts/work/_temp/temp_web_package_21475337782320203.zip
Source for image: { type: 'SharedImageVersion',
  imageVersionId:
    '/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/<galleryName>/images/<imageDefName>/versions/<imgVersionNumber>' }
template name: t_1556938436xxx
starting put template...
```

When the image build starts, the run status is reported in the release logs:

```
starting run template...
```

When the image build finishes, the output is similar to following text:

```
2019-05-06T12:49:52.0558229Z starting run template...
2019-05-06T13:36:33.8863094Z run template: Succeeded
2019-05-06T13:36:33.8867768Z getting runOutput for SharedImage_distribute
2019-05-06T13:36:34.6652541Z =====
2019-05-06T13:36:34.6652925Z ## task output variables ##
2019-05-06T13:36:34.6658728Z $(imageUri) =
/subscriptions/<subscriptionID>/resourceGroups/aibwinsig/providers/Microsoft.Compute/galleries/my22stSIG/images/winWAppimages/versions/0.23760.13763
2019-05-06T13:36:34.6659989Z =====
2019-05-06T13:36:34.6663500Z deleting template t_1557146959485...
2019-05-06T13:36:34.6673713Z deleting storage blob imagebuilder-vststask\webapp\18-1\webapp_1557146958741.zip
2019-05-06T13:36:34.9786039Z blob imagebuilder-vststask\webapp\18-1\webapp_1557146958741.zip is deleted
2019-05-06T13:38:37.4884068Z delete template: Succeeded
```

The image template and `IT_{DestinationResourceGroup}_{TemplateName}` are deleted.

You can take the `$(imageUri)` Azure DevOps Services (formerly Visual Studio Team Services, or VSTS) variable and use it in the next task or just use the value and build a VM.

## Output DevOps variables

Here are the publisher, offer, SKU, and version of the source marketplace image:

- `$(pirPublisher)`
- `$(pirOffer)`
- `$(pirSku)`
- `$(pirVersion)`

Here's the image URI, which is the resource ID of the distributed image:

- `$(imageUri)`

## FAQ

**Can I use an existing image template that I've already created, outside of DevOps?**

Not at this time.

**Can I specify the image template name?**

No. A unique template name is used and then deleted.

**The VM Image Builder task failed. How can I troubleshoot the issue?**

If there's a build failure, the DevOps task doesn't delete the staging resource group. You can access the staging resource group that contains the build customization log.

You'll see an error in the DevOps log for the VM Image Builder task, and the message will contain the `customization.log` location. For example:

Agent job		Started: 12/2/2019, 7:09:33 PM
Pool: Azure Pipelines · Agent: Hosted Agent		··· 17m 27s
✓	Initialize job · succeeded	6s
✓	Download Artifacts · succeeded	6s
✗	Azure VM Image Builder Task · 1 error	 17m 14s
✗	Error: post template call failed for template t_1575342592203 with error: Failed in building/customizing image: Failed while waiting for packerizer: Microservice has failed: Failed while processing request: Error when executing packerizer: Packer build command has failed: exit status 1. During the image build, a failure has occurred, please review the build log to identify which build/customization step failed. For more troubleshooting steps go to https://aka.ms/azvmimagebuildert . Image Build log location: https://0huje17fvia32z8xpe7e7h9i.blob.core.windows.net/packerlogs/4288328b-c206-4a7a-8085-e9502a5837a3/customization.log. OperationId: d758bb20-314a-4b81-8747-93437f3fabc8. Use this operationId to search packer logs. (CODE: 200)	

For more information, see [Troubleshoot the VM Image Builder service](#).

After you've investigated the failure, you can delete the staging resource group. First, delete the VM Image Builder template resource artifact. The artifact is prefixed with `t_`, and you can find it in the DevOps task build log:

```
...
Source for image: { type: 'SharedImageVersion',
  imageVersionId:
  '/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/<galleryName>
  /images/<imageDefName>/versions/<imgVersionNumber>' }
...
template name: t_1556938436xxx
...
```

The VM Image Builder template resource artifact is in the resource group that was specified initially in the task. When you're done troubleshooting, delete the artifact. If you're deleting it by using the Azure portal, within the resource group, select **Show Hidden Types** to view the artifact.

## Next steps

For more information, see [VM Image Builder overview](#).

# Create an Azure Image Builder Bicep or ARM JSON template

9/21/2022 • 29 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure Image Builder uses a Bicep file or an ARM JSON template file to pass information into the Image Builder service. In this article we'll go over the sections of the files, so you can build your own. For latest API versions, see [template reference](#). To see examples of full json files, see the [Azure Image Builder GitHub](#).

The basic format is:

- [JSON](#)
- [Bicep](#)

```
{  
  "type": "Microsoft.VirtualMachineImages/imageTemplates",  
  "apiVersion": "2022-02-14",  
  "location": "<region>",  
  "tags": {  
    "<name>": "<value>",  
    "<name>": "<value>"  
  },  
  "identity": {},  
  "properties": {  
    "buildTimeoutInMinutes": <minutes>,  
    "customize": [],  
    "distribute": [],  
    "source": {},  
    "stagingResourceGroup": "/subscriptions/<subscriptionID>/resourceGroups/<stagingResourceGroupName>",  
    "validate": {},  
    "vmProfile": {  
      "vmSize": "<vmSize>",  
      "proxyVmSize": "<vmSize>",  
      "osDiskSizeGB": <sizeInGB>,  
      "vnetConfig": {  
        "subnetId": "  
          "/subscriptions/<subscriptionID>/resourceGroups/<vnetRgName>/providers/Microsoft.Network/virtualNetworks/<vn  
          etName>/subnets/<subnetName>"  
        },  
        "userAssignedIdentities": [  
  
          "/subscriptions/<subscriptionID>/resourceGroups/<identityRgName>/providers/Microsoft.ManagedIdentity/userAss  
          ignedIdentities/<identityName1>",  
  
          "/subscriptions/<subscriptionID>/resourceGroups/<identityRgName>/providers/Microsoft.ManagedIdentity/userAss  
          ignedIdentities/<identityName2>",  
  
          "/subscriptions/<subscriptionID>/resourceGroups/<identityRgName>/providers/Microsoft.ManagedIdentity/userAss  
          ignedIdentities/<identityName3>",  
          ...  
        ]  
      }  
    }  
  }  
}
```

## Type and API version

The `type` is the resource type, which must be `Microsoft.VirtualMachineImages/imageTemplates`. The `apiVersion` will change over time as the API changes. See [What's new in Azure VM Image Builder](#) for all major API changes and feature updates for the Azure VM Image Builder service.

- [JSON](#)
- [Bicep](#)

```
"type": "Microsoft.VirtualMachineImages/imageTemplates",  
"apiVersion": "2022-02-14",
```

## Location

The location is the region where the custom image will be created. The following regions are supported:

- East US

- East US 2
- West Central US
- West US
- West US 2
- West US 3
- South Central US
- North Europe
- West Europe
- South East Asia
- Australia Southeast
- Australia East
- UK South
- UK West
- Brazil South
- Canada Central
- Central India
- Central US
- France Central
- Germany West Central
- Japan East
- North Central US
- Norway East
- Switzerland North
- Jio India West
- UAE North
- East Asia
- Korea Central
- South Africa North
- USGov Arizona (Public Preview)
- USGov Virginia (Public Preview)

#### **IMPORTANT**

Register the feature `Microsoft.VirtualMachineImages/FairfaxPublicPreview` to access the Azure Image Builder public preview in Azure Government regions (USGov Arizona and USGov Virginia).

Use the following command to register the feature for Azure Image Builder in Azure Government regions (USGov Arizona and USGov Virginia).

- [Azure PowerShell](#)
- [Azure CLI](#)

```
Register-AzProviderPreviewFeature -ProviderNamespace Microsoft.VirtualMachineImages -Name FairfaxPublicPreview
```

- [JSON](#)
- [Bicep](#)

```
"location": "<region>"
```

#### **Data residency**

The Azure VM Image Builder service doesn't store or process customer data outside regions that have strict single region data residency requirements when a customer requests a build in that region. If a service outage for regions that have data residency requirements, you'll need to create Bicep files/templates in a different region and geography.

#### **Zone redundancy**

Distribution supports zone redundancy, VHDs are distributed to a Zone Redundant Storage (ZRS) account by default and the Azure Compute Gallery (formerly known as Shared Image Gallery) version will support a [ZRS storage type](#) if specified.

## Tags

Tags are key/value pairs you can specify for the image that's generated.

## Identity

There are two ways to add user assigned identities explained below.

### User-assigned identity for Azure Image Builder image template resource

Required - For Image Builder to have permissions to read/write images, and read in scripts from Azure Storage, you must create an Azure user-assigned identity that has permissions to the individual resources. For details on how Image Builder permissions work, and relevant steps, see [Create an image and use a user-assigned managed identity to access files in an Azure storage account](#).

- [JSON](#)
- [Bicep](#)

```
"identity": {  
    "type": "UserAssigned",  
    "userAssignedIdentities": {  
        "<imgBuilderId>": {}  
    }  
}
```

The Image Builder service User Assigned Identity:

- Supports a single identity only.
- Doesn't support custom domain names.

To learn more, see [What is managed identities for Azure resources?](#). For more information on deploying this feature, see [Configure managed identities for Azure resources on an Azure VM using Azure CLI](#).

### User-assigned identity for the Image Builder Build VM

This property is only available in API versions [2021-10-01](#) or newer.

Optional - The Image Builder Build VM, that is created by the Image Builder service in your subscription, is used to build and customize the image. For the Image Builder Build VM to have permissions to authenticate with other services like Azure Key Vault in your subscription, you must create one or more Azure User Assigned Identities that have permissions to the individual resources. Azure Image Builder can then associate these User Assigned Identities with the Build VM. Customizer scripts running inside the Build VM can then fetch tokens for these identities and interact with other Azure resources as needed. Be aware, the user assigned identity for Azure Image Builder must have the "Managed Identity Operator" role assignment on all the user assigned identities for Azure Image Builder to be able to associate them to the build VM.

#### NOTE

Be aware that multiple identities can be specified for the Image Builder Build VM, including the identity you created for the [image template resource](#). By default, the identity you created for the image template resource won't automatically be added to the build VM.

- [JSON](#)
- [Bicep](#)

```
"properties": {  
    "vmProfile": {  
        "userAssignedIdentities": [  
  
            "/subscriptions/<subscriptionID>/resourceGroups/<identityRgName>/providers/Microsoft.ManagedIdentity/userAss  
ignedIdentities/<identityName>"  
        ]  
    }  
}
```

The Image Builder Build VM User Assigned Identity:

- Supports a list of one or more user assigned managed identities to be configured on the VM.
- Supports cross subscription scenarios (identity created in one subscription while the image template is created in another subscription under the same tenant).
- Doesn't support cross tenant scenarios (identity created in one tenant while the image template is created in another tenant).

To learn more, see:

- [How to use managed identities for Azure resources on an Azure VM to acquire an access token](#)
- [How to use managed identities for Azure resources on an Azure VM for sign-in](#)

## Properties: buildTimeoutInMinutes

Maximum duration to wait while building the image template (includes all customizations, validations, and distributions).

If you don't specify the property or set the value to 0, the default value is used, which is 240 minutes or four hours. The minimum value is 6 minutes, and the maximum value is 960 minutes or 16 hours. When the timeout value is hit (whether or not the image build is complete), you'll see an error similar to:

```
[ERROR] Failed while waiting for packerizer: Timeout waiting for microservice to  
[ERROR] complete: 'context deadline exceeded'
```

For Windows, we don't recommend setting `buildTimeoutInMinutes` below 60 minutes. If you find you're hitting the timeout, review the [logs](#) to see if the customization step is waiting on something like user input. If you find you need more time for customizations to complete, increase the `buildTimeoutInMinutes` value. But, don't set it too high because you might have to wait for it to time out before seeing an error.

## Properties: customize

Image Builder supports multiple "customizers", which are functions used to customize your image, such as running scripts, or rebooting servers.

When using `customize`:

- You can use multiple customizers.
- Customizers execute in the order specified in the template.
- If one customizer fails, then the whole customization component will fail and report back an error.
- Test the scripts thoroughly before using them in a template. Debugging the scripts by themselves is easier.
- Don't put sensitive data in the scripts. Inline commands can be viewed in the image template definition. If you have sensitive information (including passwords, SAS token, authentication tokens, etc.), it should be moved into scripts in Azure Storage, where access requires authentication.
- The script locations need to be publicly accessible, unless you're using [MSI](#).

The `customize` section is an array. The supported customizer types are: File, PowerShell, Shell, WindowsRestart, and WindowsUpdate.

- [JSON](#)
- [Bicep](#)

```
"customize": [  
    {  
        "type": "File",  
        "destination": "string",  
        "sha256Checksum": "string",  
        "sourceUri": "string"  
    },  
    {  
        "type": "PowerShell",  
        "inline": [ "string" ],  
        "runAsSystem": "bool",  
        "runElevated": "bool",  
        "scriptUri": "string",  
        "sha256Checksum": "string",  
        "validExitCodes": [ "int" ]  
    },  
    {  
        "type": "Shell",  
        "inline": [ "string" ],  
        "scriptUri": "string",  
        "sha256Checksum": "string"  
    },  
    {  
        "type": "WindowsRestart",  
        "restartCheckCommand": "string",  
        "restartCommand": "string",  
        "restartTimeout": "string"  
    },  
    {  
        "type": "WindowsUpdate",  
        "filters": [ "string" ],  
        "searchCriteria": "string",  
        "updateLimit": "int"  
    }  
]
```

### Shell customizer

The `Shell` customizer supports running shell scripts on Linux. The shell scripts must be publicly accessible or

you must have configured an [MSI](#) for Image Builder to access them.

- [JSON](#)
- [Bicep](#)

```
"customize": [
  {
    "type": "Shell",
    "name": "<name>",
    "scriptUri": "<link to script>",
    "sha256Checksum": "<sha256 checksum>"
  }
],
"customize": [
  {
    "type": "Shell",
    "name": "<name>",
    "inline": "<commands to run>"
  }
]
```

Customize properties:

- **type** – Shell.
- **name** - name for tracking the customization.
- **scriptUri** - URI to the location of the file.
- **inline** - array of shell commands, separated by commas.
- **sha256Checksum** - Value of sha256 checksum of the file, you generate this value locally, and then Image Builder will checksum and validate.

To generate the sha256Checksum, using a terminal on Mac/Linux run: `sha256sum <fileName>`

#### NOTE

Inline commands are stored as part of the image template definition, you can see these when you dump out the image definition. If you have sensitive commands or values (including passwords, SAS token, authentication tokens etc), it's recommended these are moved into scripts, and use a user identity to authenticate to Azure Storage.

#### Super user privileges

Prefix the commands with `sudo` to run them with super user privileges. You can add the commands into scripts or use it inline commands, for example:

- [JSON](#)
- [Bicep](#)

```
"type": "Shell",
"name": "setupBuildPath",
"inline": [
  "sudo mkdir /buildArtifacts",
  "sudo cp /tmp/index.html /buildArtifacts/index.html"
]
```

Example of a script using sudo that you can reference using scriptUri:

```
#!/bin/bash -e

echo "Telemetry: creating files"
mkdir /myfiles

echo "Telemetry: running sudo 'as-is' in a script"
sudo touch /myfiles/somethingElevated.txt
```

#### Windows restart customizer

The `WindowsRestart` customizer allows you to restart a Windows VM and wait for the VM come back online, this customizer allows you to install software that requires a reboot.

- [JSON](#)
- [Bicep](#)

```

"customize": [
  {
    "type": "WindowsRestart",
    "restartCommand": "shutdown /r /f /t 0",
    "restartCheckCommand": "echo Azure-Image-Builder-Restarted-the-VM >
c:\\buildArtifacts\\azureImageBuilderRestart.txt",
    "restartTimeout": "5m"
  }
]

```

Customize properties:

- **Type**: WindowsRestart.
- **restartCommand** - Command to execute the restart (optional). The default is `'shutdown /r /f /t 0 /c \"packer restart\"'`.
- **restartCheckCommand** – Command to check if restart succeeded (optional).
- **restartTimeout** - Restart timeout specified as a string of magnitude and unit. For example, `5m` (5 minutes) or `2h` (2 hours). The default is: `5m`.

#### NOTE

There's no Linux restart customizer. If you're installing drivers, or components that require a restart, you can install them and invoke a restart using the Shell customizer. There's a 20min SSH timeout to the build VM.

### PowerShell customizer

The `PowerShell` customizer supports running PowerShell scripts and inline command on Windows, the scripts must be publicly accessible for the IB to access them.

- [JSON](#)
- [Bicep](#)

```

"customize": [
  {
    "type": "PowerShell",
    "name": "<name>",
    "scriptUri": "<path to script>",
    "runElevated": <true false>,
    "sha256Checksum": "<sha256 checksum>"
  },
  {
    "type": "PowerShell",
    "name": "<name>",
    "inline": "<PowerShell syntax to run>",
    "validExitCodes": [<exit code>],
    "runElevated": <true or false>
  }
]

```

Customize properties:

- **type** – PowerShell.
- **scriptUri** - URI to the location of the PowerShell script file.
- **inline** – Inline commands to be run, separated by commas.
- **validExitCodes** – Optional, valid codes that can be returned from the script/inline command. The `runElevated` property avoids reported failure of the script/inline command.
- **runElevated** – Optional, boolean, support for running commands and scripts with elevated permissions.
- **sha256Checksum** - generate the SHA256 checksum of the file locally, update the checksum value to lowercase, and Image Builder will validate the checksum during the deployment of the image template.

To generate the sha256Checksum, use the [Get-FileHash](#) cmdlet in PowerShell.

### File customizer

The `File` customizer lets Image Builder download a file from a GitHub repo or Azure storage. The customizer supports both Linux and Windows. If you have an image build pipeline that relies on build artifacts, you can set the file customizer to download from the build share, and move the artifacts into the image.

- [JSON](#)
- [Bicep](#)

```

"customize": [
  {
    "type": "File",
    "name": "<name>",
    "sourceUri": "<source location>",
    "destination": "<destination>",
    "sha256Checksum": "<sha256 checksum>"
  }
]

```

File customizer properties:

- **sourceUri** - an accessible storage endpoint, this endpoint can be GitHub or Azure storage. You can only download one file, not an entire directory. If you need to download a directory, use a compressed file, then uncompress it using the Shell or PowerShell customizers.

#### NOTE

If the sourceUri is an Azure Storage Account, irrespective if the blob is marked public, you'll to grant the Managed User Identity permissions to read access on the blob. See this [example](#) to set the storage permissions.

- **destination** – the full destination path and file name. Any referenced path and subdirectories must exist, use the Shell or PowerShell customizers to set up these paths up beforehand. You can use the script customizers to create the path.

This customizer is supported by Windows directories and Linux paths, but there are some differences:

- Linux – the only path Image builder can write to is /tmp.
- Windows – No path restriction, but the path must exist.

If there's an error trying to download the file, or put it in a specified directory, then customize step will fail, and this error will be in the customization.log.

#### NOTE

The file customizer is only suitable for small file downloads, < 20MB. For larger file downloads, use a script or inline command, then use code to download files, such as, Linux `wget` or `curl`, Windows, `Invoke-WebRequest`.

- **sha256Checksum** - generate the SHA256 checksum of the file locally, update the checksum value to lowercase, and Image Builder will validate the checksum during the deployment of the image template.

To generate the sha256Checksum, use the `Get-FileHash` cmdlet in PowerShell.

### Windows update customizer

The `WindowsUpdate` customizer is built on the [community Windows Update Provisioner](#) for Packer, which is an open source project maintained by the Packer community. Microsoft tests and validate the provisioner with the Image Builder service, and will support investigating issues with it, and work to resolve issues, however the open source project isn't officially supported by Microsoft. For detailed documentation on and help with the Windows Update Provisioner, see the project repository.

- **JSON**
- **Bicep**

```

"customize": [
  {
    "type": "WindowsUpdate",
    "searchCriteria": "IsInstalled=0",
    "filters": [
      "exclude:$_.Title -like '*Preview*'",
      "include:$true"
    ],
    "updateLimit": 20
  }
]

```

Customizer properties:

- **type** – WindowsUpdate.
- **searchCriteria** - Optional, defines which type of updates are installed (like Recommended or Important), BrowseOnly=0 and IsInstalled=0 (Recommended) is the default.
- **filters** – Optional, allows you to specify a filter to include or exclude updates.
- **updateLimit** – Optional, defines how many updates can be installed, default 1000.

#### NOTE

The Windows Update customizer can fail if there are any outstanding Windows restarts, or application installations still running, typically you may see this error in the customization.log,

`System.Runtime.InteropServices.COMException (0x80240016): Exception from HRESULT: 0x80240016`. We strongly advise you consider adding in a Windows Restart, and/or allowing applications enough time to complete their installations using `sleep` or `wait` commands in the inline commands or scripts before running Windows Update.

#### Generalize

By default, Azure Image Builder will also run `deprovision` code at the end of each image customization phase, to generalize the image. Generalizing is a process where the image is set up so it can be reused to create multiple VMs. For Windows VMs, Azure Image Builder uses Sysprep. For Linux, Azure Image Builder runs `waagent -deprovision`.

The commands Image Builder users to generalize may not be suitable for every situation, so Azure Image Builder will allow you to customize this command, if needed.

If you're migrating existing customization, and you're using different Sysprep/waagent commands, you can use the Image Builder generic commands, and if the VM creation fails, use your own Sysprep or waagent commands.

If Azure Image Builder creates a Windows custom image successfully, and you create a VM from it, then find that the VM creation fails or doesn't complete successfully, you'll need to review the Windows Server Sysprep documentation or raise a support request with the Windows Server Sysprep Customer Services Support team, who can troubleshoot and advise on the correct Sysprep usage.

#### Default Sysprep command

```
Write-Output '">>>> Waiting for GA Service (RdAgent) to start ...'
while ((Get-Service RdAgent).Status -ne 'Running') { Start-Sleep -s 5 }
Write-Output '">>>> Waiting for GA Service (WindowsAzureTelemetryService) to start ...'
while ((Get-Service WindowsAzureTelemetryService) -and ((Get-Service WindowsAzureTelemetryService).Status -ne 'Running')) { Start-Sleep -s 5 }
Write-Output '">>>> Waiting for GA Service (WindowsAzureGuestAgent) to start ...'
while ((Get-Service WindowsAzureGuestAgent).Status -ne 'Running') { Start-Sleep -s 5 }
Write-Output '">>>> Sysprepping VM ...'
if( Test-Path $Env:SystemRoot\system32\Sysprep\unattend.xml ) {
    Remove-Item $Env:SystemRoot\system32\Sysprep\unattend.xml -Force
}
& $Env:SystemRoot\System32\Sysprep\Sysprep.exe /oobe /generalize /quiet /quit
while($true) {
    $imageState = (Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup\State).ImageState
    Write-Output $imageState
    if ($imageState -eq 'IMAGE_STATE_GENERALIZE_RESEAL_TO_OOBE') { break }
    Start-Sleep -s 5
}
Write-Output '">>>> Sysprep complete ...'
```

#### Default Linux deprovision command

```
WAAGENT=/usr/sbin/waagent
waagent -version 1> /dev/null 2>&1
if [ $? -eq 0 ]; then
    WAAGENT=waagent
fi
$WAAGENT -force -deprovision+user && export HISTSIZE=0 && sync
```

#### Overriding the Commands

To override the commands, use the PowerShell or Shell script provisioners to create the command files with the exact file name, and put them in the correct directories:

- Windows: c:\DeprovisioningScript.ps1
- Linux: /tmp/DeprovisioningScript.sh

Image Builder will read these commands, these commands are written out to the AIB logs, `customization.log`. See [troubleshooting](#) on how to collect logs.

## Properties: distribute

Azure Image Builder supports three distribution targets:

- **managedImage** - managed image.
- **sharedImage** - Azure Compute Gallery.
- **VHD** - VHD in a storage account.

You can distribute an image to both of the target types in the same configuration.

**NOTE**

The default AIB sysprep command doesn't include "/mode:vm", however this property maybe required when create images that will have the HyperV role installed. If you need to add this command argument, you must override the sysprep command.

Because you can have more than one target to distribute to, Image Builder maintains a state for every distribution target that can be accessed by querying the `runOutputName`. The `runOutputName` is an object you can query post distribution for information about that distribution. For example, you can query the location of the VHD, or regions where the image version was replicated to, or SIG Image version created. This is a property of every distribution target. The `runOutputName` must be unique to each distribution target. Here's an example for querying an Azure Compute Gallery distribution:

```
subscriptionID=<subscriptionID>
imageResourceGroup=<resourceGroup of image template>
runOutputName=<runOutputName>

az resource show \
--ids
"/subscriptions/$subscriptionID/resourcegroups/$imageResourceGroup/providers/Microsoft.VirtualMachineImages/
imageTemplates/ImageTemplateLinuxRHEL77/runOutputs/$runOutputName" \
--api-version=2021-10-01
```

Output:

```
{
  "id": "/subscriptions/xxxxxx/resourcegroups/rheltest/providers/Microsoft.VirtualMachineImages/imageTemplates/ImageTemplateLinuxRHEL77/runOutputs/rhel77",
  "identity": null,
  "kind": null,
  "location": null,
  "managedBy": null,
  "name": "rhel77",
  "plan": null,
  "properties": {
    "artifactId": "/subscriptions/xxxxxx/resourceGroups/aibDevOpsImg/providers/Microsoft.Compute/galleries/devOpsSIG/images/rhel/versions/0.24105.52755",
    "provisioningState": "Succeeded"
  },
  "resourceGroup": "rheltest",
  "sku": null,
  "tags": null,
  "type": "Microsoft.VirtualMachineImages/imageTemplates/runOutputs"
}
```

**Distribute: managedImage**

The image output will be a managed image resource.

- [JSON](#)
- [Bicep](#)

```
{
  "type": "managedImage",
  "imageId": "<resource ID>",
  "location": "<region>",
  "runOutputName": "<name>",
  "artifactTags": {
    "<name>": "<value>",
    "<name>": "<value>"
  }
}
```

Distribute properties:

- **type** – managedImage
- **imageId** – Resource ID of the destination image, expected format:  
`/subscriptions/<subscriptionId>/resourceGroups/<destinationResourceGroupName>/providers/Microsoft.Compute/images/<imageName>`
- **location** - location of the managed image.
- **runOutputName** – unique name for identifying the distribution.
- **artifactTags** - Optional user specified key\value tags.

#### NOTE

The destination resource group must exist. If you want the image distributed to a different region, it will increase the deployment time.

#### Distribute: sharedImage

The Azure Compute Gallery is a new Image Management service that allows managing of image region replication, versioning and sharing custom images. Azure Image Builder supports distributing with this service, so you can distribute images to regions supported by Azure Compute Galleries.

An Azure Compute Gallery is made up of:

- **Gallery** - Container for multiple images. A gallery is deployed in one region.
- **Image definitions** - a conceptual grouping for images.
- **Image versions** - an image type used for deploying a VM or scale set. Image versions can be replicated to other regions where VMs need to be deployed.

Before you can distribute to the gallery, you must create a gallery and an image definition, see [Create a gallery](#).

- [JSON](#)
- [Bicep](#)

```
{  
    "type": "SharedImage",  
    "galleryImageId": "<resource ID>",  
    "runOutputName": "<name>",  
    "artifactTags": {  
        "<name>": "<value>",  
        "<name>": "<value>"  
    },  
    "replicationRegions": [  
        "<region where the gallery is deployed>",  
        "<region>"  
    ]  
}
```

Distribute properties for galleries:

- **type** - sharedImage
- **galleryImageId** – ID of the Azure Compute Gallery, this property can be specified in two formats:
  - Automatic versioning - Image Builder will generate a monotonic version number for you, this property is useful for when you want to keep rebuilding images from the same template: The format is:  
`/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/galleries/<sharedImageGalleryName>/image`
  - Explicit versioning - You can pass in the version number you want image builder to use. The format is:  
`/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/<sharedImageGalName>/images/<imageDefName>` - for example: 1.1.1
- **runOutputName** – unique name for identifying the distribution.
- **artifactTags** - optional user specified key\value tags.
- **replicationRegions** - array of regions for replication. One of the regions must be the region where the Gallery is deployed. Adding regions will mean an increase of build time, as the build doesn't complete until the replication has completed.
- **excludeFromLatest** (optional) - allows you to mark the image version you create not be used as the latest version in the gallery definition, the default is 'false'.
- **storageAccountType** (optional) - AIB supports specifying these types of storage for the image version that is to be created:
  - "Standard\_LRS"
  - "Standard\_ZRS"","

#### NOTE

If the image template and referenced `image definition` aren't in the same location, you'll see additional time to create images. Image Builder currently doesn't have a `location` parameter for the image version resource, we take it from its parent `image definition`. For example, if an image definition is in `westus` and you want the image version replicated to `eastus`, a blob is copied to `westus`, an image version resource in `westus` is created, and then replicate to `eastus`. To avoid the additional replication time, ensure the `image definition` and image template are in the same location.

### Distribute: VHD

You can output to a VHD. You can then copy the VHD, and use it to publish to Azure MarketPlace, or use with Azure Stack.

- [JSON](#)
- [Bicep](#)

```
{  
  "type": "VHD",  
  "runOutputName": "<VHD name>",  
  "artifactTags": {  
    "<name>": "<value>",  
    "<name>": "<value>"  
  }  
}
```

OS Support: Windows and Linux

Distribute VHD parameters:

- **type** - VHD.
- **runOutputName** – unique name for identifying the distribution.
- **tags** - Optional user specified key value pair tags.

Azure Image Builder doesn't allow the user to specify a storage account location, but you can query the status of the `runOutputs` to get the location.

```
az resource show \  
  --ids  
  "/subscriptions/$subscriptionId/resourcegroups/<imageResourceGroup>/providers/Microsoft.VirtualMachineImages  
  /imageTemplates/<imageTemplateName>/runOutputs/<runOutputName>" | grep artifactUri
```

#### NOTE

Once the VHD has been created, copy it to a different location, as soon as possible. The VHD is stored in a storage account in the temporary resource group created when the image template is submitted to the Azure Image Builder service. If you delete the image template, then you'll lose the VHD.

## Properties: source

The `source` section contains information about the source image that will be used by Image Builder.

The API requires a `SourceType` that defines the source for the image build, currently there are three types:

- **PlatformImage** - indicated the source image is a Marketplace image.
- **ManagedImage** - used when starting from a regular managed image.
- **SharedImageVersion** - used when you're using an image version in an Azure Compute Gallery as the source.

#### NOTE

When using existing Windows custom images, you can run the Sysprep command up to three times on a single Windows 7 or Windows Server 2008 R2 image, or 1001 times on a single Windows image for later versions; for more information, see the [sysprep](#) documentation.

### PlatformImage source

Azure Image Builder supports Windows Server and client, and Linux Azure Marketplace images, see [Learn about Azure Image Builder](#) for the full list.

- [JSON](#)
- [Bicep](#)

```
"source": {  
  "type": "PlatformImage",  
  "publisher": "Canonical",  
  "offer": "UbuntuServer",  
  "sku": "18.04-LTS",  
  "version": "latest"  
}
```

The properties here are the same that are used to create VM's, using AZ CLI, run the below to get the properties:

```
az vm image list -l westus -f UbuntuServer -p Canonical --output table --all
```

You can use `latest` in the version, the version is evaluated when the image build takes place, not when the template is submitted. If you use this functionality with the Azure Compute Gallery destination, you can avoid resubmitting the template, and rerun the image build at intervals, so your images are recreated from the most recent images.

#### Support for market place plan information

You can also specify plan information, for example:

- [JSON](#)
- [Bicep](#)

```
"source": {  
    "type": "PlatformImage",  
    "publisher": "RedHat",  
    "offer": "rhel-byos",  
    "sku": "rhel-lvm75",  
    "version": "latest",  
    "planInfo": {  
        "planName": "rhel-lvm75",  
        "planProduct": "rhel-byos",  
        "planPublisher": "redhat"  
    }  
}
```

#### ManagedImage source

Sets the source image as an existing managed image of a generalized VHD or VM.

##### NOTE

The source managed image must be of a supported OS and the image must reside in the same subscription and region as your Azure Image Builder template.

- [JSON](#)
- [Bicep](#)

```
"source": {  
    "type": "ManagedImage",  
    "imageId":  
    "/subscriptions/<subscriptionId>/resourceGroups/{destinationResourceGroupName}/providers/Microsoft.Compute/i  
mages/<imageName>"  
}
```

The `imageId` should be the ResourceId of the managed image. Use `az image list` to list available images.

#### SharedImageVersion source

Sets the source image as an existing image version in an Azure Compute Gallery.

##### NOTE

The source shared image version must be of a supported OS and the image version must reside in the same region as your Azure Image Builder template, if not, replicate the image version to the Image Builder Template region.

- [JSON](#)
- [Bicep](#)

```
"source": {  
    "type": "SharedImageVersion",  
    "imageVersionId":  
    "/subscriptions/<subscriptionId>/resourceGroups/<resourceGroup>/providers/Microsoft.Compute/galleries/<share  
dImageGalleryName>/images/<imageDefinitionName>/versions/<imageVersion>"  
}
```

The `imageVersionId` should be the ResourceId of the image version. Use `az sig image-version list` to list image versions.

## Properties: stagingResourceGroup

The `stagingResourceGroup` property contains information about the staging resource group that the Image

Builder service will create for use during the image build process. The `stagingResourceGroup` is an optional property for anyone who wants more control over the resource group created by Image Builder during the image build process. You can create your own resource group and specify it in the `stagingResourceGroup` section or have Image Builder create one on your behalf.

- [JSON](#)
- [Bicep](#)

```
"properties": {  
    "stagingResourceGroup": "/subscriptions/<subscriptionID>/resourceGroups/<stagingResourceGroupName>"  
}
```

#### Template creation scenarios

- The `stagingResourceGroup` property is left empty

If the `stagingResourceGroup` property isn't specified or specified with an empty string, the Image Builder service will create a staging resource group with the default name convention "IT\_\*\*\*". The staging resource group will have the default tags applied to it: `createdBy`, `imageTemplateName`, `imageTemplateResourceGroupName`. Also, the default RBAC will be applied to the identity assigned to the Azure Image Builder template resource, which is "Contributor".

- The `stagingResourceGroup` property is specified with a resource group that exists

If the `stagingResourceGroup` property is specified with a resource group that does exist, then the Image Builder service will check to make sure the resource group isn't associated with another image template, is empty (no resources inside), in the same region as the image template, and has either "Contributor" or "Owner" RBAC applied to the identity assigned to the Azure Image Builder image template resource. If any of the aforementioned requirements aren't met, an error will be thrown. The staging resource group will have the following tags added to it: `usedBy`, `imageTemplateName`, `imageTemplateResourceGroupName`. Pre-existing tags aren't deleted.

- The `stagingResourceGroup` property is specified with a resource group that doesn't exist

If the `stagingResourceGroup` property is specified with a resource group that doesn't exist, then the Image Builder service will create a staging resource group with the name provided in the `stagingResourceGroup` property. There will be an error if the given name doesn't meet Azure naming requirements for resource groups. The staging resource group will have the default tags applied to it: `createdBy`, `imageTemplateName`, `imageTemplateResourceGroupName`. By default the identity assigned to the Azure Image Builder image template resource will have the "Contributor" RBAC applied to it in the resource group.

#### Template deletion

Any staging resource group created by the Image Builder service will be deleted after the image template is deleted. The deletion includes staging resource groups that were specified in the `stagingResourceGroup` property, but didn't exist prior to the image build.

If Image Builder didn't create the staging resource group, but the resources inside of the resource group, those resources will be deleted after the image template is deleted, given the Image Builder service has the appropriate permissions or role required to delete resources.

## Properties: validate

You can use the `validate` property to validate platform images and any customized images you create regardless of if you used Azure Image Builder to create them.

Azure Image Builder supports a 'Source-Validation-Only' mode that can be set using the `sourceValidationOnly` property. If the `sourceValidationOnly` property is set to true, the image specified in the `source` section will directly be validated. No separate build will be run to generate and then validate a customized image.

The `inVMValidations` property takes a list of validators that will be performed on the image. Azure Image Builder supports both PowerShell and Shell validators.

The `continueDistribute onFailure` property is responsible for whether the output image(s) will be distributed if validation fails. By default, if validation fails and this property is set to false, the output image(s) won't be distributed. If validation fails and this property is set to true, the output image(s) will still be distributed. Use this option with caution as it may result in failed images being distributed for use. In either case (true or false), the end to end image run will be reported as a failed if a validation failure. This property has no effect on whether validation succeeds or not.

When using `validate`:

- You can use multiple validators.
- Validators execute in the order specified in the template.
- If one validator fails, then the whole validation component will fail and report back an error.

- It's advised you test the script thoroughly before using it in a template. Debugging the script on your own VM will be easier.
- Don't put sensitive data in the scripts.
- The script locations need to be publicly accessible, unless you're using [MSI](#).

How to use the `validate` property to validate Windows images:

- [JSON](#)
- [Bicep](#)

```
{
  "properties": {
    "validate": [
      {
        "continueDistributeOnFailure": false,
        "sourceValidationOnly": false,
        "inVMValidations": [
          {
            "type": "PowerShell",
            "name": "test PowerShell validator inline",
            "inline": [
              "<command to run inline>"
            ],
            "validExitCodes": <exit code>,
            "runElevated": <true or false>,
            "runAsSystem": <true or false>
          },
          {
            "type": "PowerShell",
            "name": "<name>",
            "scriptUri": "<path to script>",
            "runElevated": <true false>,
            "sha256Checksum": "<sha256 checksum>"
          }
        ]
      }
    ]
  }
}
```

`inVMValidations` properties:

- **type** – PowerShell.
- **name** - name of the validator
- **scriptUri** - URI of the PowerShell script file.
- **inline** – array of commands to be run, separated by commas.
- **validExitCodes** – Optional, valid codes that can be returned from the script/inline command, this will avoid reported failure of the script/inline command.
- **runElevated** – Optional, boolean, support for running commands and scripts with elevated permissions.
- **sha256Checksum** - Value of sha256 checksum of the file, you generate this locally, and then Image Builder will checksum and validate.

To generate the sha256Checksum, using a PowerShell on Windows [Get-Hash](#)

How to use the `validate` property to validate Linux images:

- [JSON](#)
- [Bicep](#)

```
{
  "properties": {
    "validate": {
      "continueDistributeOnFailure": false,
      "sourceValidationOnly": false,
      "inVMValidations": [
        {
          "type": "Shell",
          "name": "<name>",
          "inline": [
            "<command to run inline>"
          ]
        },
        {
          "type": "Shell",
          "name": "<name>",
          "scriptUri": "<path to script>",
          "sha256Checksum": "<sha256 checksum>"
        }
      ]
    }
  }
}
```

`inVMValidations` properties:

- **type** – Shell
- **name** - name of the validator
- **scriptUri** - URI of the script file
- **inline** - array of commands to be run, separated by commas.
- **sha256Checksum** - Value of sha256 checksum of the file, you generate this locally, and then Image Builder will checksum and validate.

To generate the sha256Checksum, using a terminal on Mac/Linux run: `sha256sum <fileName>`

## Properties: vmProfile

### vmSize (optional)

Image Builder uses a default SKU size of `Standard_D1_v2` for Gen1 images and `Standard_D2ds_v4` for Gen2 images. The generation is defined by the image you specify in the `source`. You can override `vmSize` for these reasons:

- Performing customizations that require increased memory, CPU and handling large files (GBs).
- Running Windows builds, you should use "Standard\_D2\_v2" or equivalent VM size.
- Require [VM isolation](#).
- Customize an image that requires specific hardware. For example, for a GPU VM, you need a GPU VM size.
- Require end to end encryption at rest of the build VM, you need to specify the support build [VM size](#) that don't use local temporary disks.

### osDiskSizeGB

By default, Image Builder doesn't change the size of the image, it uses the size from the source image. You can optionally **only** increase the size of the OS Disk (Win and Linux), and a value of 0 means leaving the same size as the source image. You can't reduce the OS Disk size to smaller than the size from the source image.

- [JSON](#)
- [Bicep](#)

```
{
  "osDiskSizeGB": 100
}
```

### vnetConfig (optional)

If you don't specify any VNet properties, Image Builder will create its own VNet, Public IP, and network security group (NSG). The Public IP is used for the service to communicate with the build VM. If you don't want to have a Public IP or you want Image Builder to have access to your existing VNet resources, such as configuration servers (DSC, Chef, Puppet, Ansible), file shares, then you can specify a VNet. For more information, review the [networking documentation](#).

- [JSON](#)
- [Bicep](#)

```
"vnetConfig": {  
    "subnetId": "  
"/subscriptions/<subscriptionID>/resourceGroups/<vnetRgName>/providers/Microsoft.Network/virtualNetworks/<vn  
etName>/subnets/<subnetName>"  
}
```

## Image Template Operations

### Starting an Image Build

To start a build, you need to invoke 'Run' on the Image Template resource, examples of `run` commands:

```
Invoke-AzResourceAction -ResourceName $imageTemplateName -ResourceGroupName $imageResourceGroup -  
ResourceType Microsoft.VirtualMachineImages/imageTemplates -ApiVersion "2021-10-01" -Action Run -Force
```

```
az resource invoke-action \  
--resource-group $imageResourceGroup \  
--resource-type Microsoft.VirtualMachineImages/imageTemplates \  
-n helloImageTemplateLinux01 \  
--action Run
```

### Cancelling an Image Build

If you're running an image build that you believe is incorrect, waiting for user input, or you feel will never complete successfully, then you can cancel the build.

The build can be canceled anytime. If the distribution phase has started you can still cancel, but you'll need to clean up any images that may not be completed. The cancel command doesn't wait for cancel to complete, monitor `lastrunstatus.runstate` for canceling progress, using these status [commands](#).

Examples of `cancel` commands:

```
Invoke-AzResourceAction -ResourceName $imageTemplateName -ResourceGroupName $imageResourceGroup -  
ResourceType Microsoft.VirtualMachineImages/imageTemplates -ApiVersion "2021-10-01" -Action Cancel -Force
```

```
az resource invoke-action \  
--resource-group $imageResourceGroup \  
--resource-type Microsoft.VirtualMachineImages/imageTemplates \  
-n helloImageTemplateLinux01 \  
--action Cancel
```

## Next steps

There are sample json files for different scenarios in the [Azure Image Builder GitHub](#).

# Create a Linux image and distribute it to an Azure Compute Gallery

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how you can use the Azure Image Builder, and the Azure CLI, to create an image version in an [Azure Compute Gallery](#) (formerly known as Shared Image Gallery), then distribute the image globally. You can also do this using [Azure PowerShell](#).

We will be using a sample json template to configure the image. The json file we are using is here: [helloImageTemplateforSIG.json](#).

To distribute the image to an Azure Compute Gallery, the template uses `sharedImage` as the value for the `distribute` section of the template.

## Register the features

To use Azure Image Builder, you need to register the feature.

Check your registration.

```
az provider show -n Microsoft.VirtualMachineImages | grep registrationState  
az provider show -n Microsoft.KeyVault | grep registrationState  
az provider show -n Microsoft.Compute | grep registrationState  
az provider show -n Microsoft.Storage | grep registrationState  
az provider show -n Microsoft.Network | grep registrationState
```

If they do not say registered, run the following:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Set variables and permissions

We will be using some pieces of information repeatedly, so we will create some variables to store that information.

Image Builder only supports creating custom images in the same Resource Group as the source managed image. Update the resource group name in this example to be the same resource group as your source managed image.

```
# Resource group name - we are using ibLinuxGalleryRG in this example
sigResourceGroup=ibLinuxGalleryRG
# Datacenter location - we are using West US 2 in this example
location=westus2
# Additional region to replicate the image to - we are using East US in this example
additionalRegion=eastus
# name of the Azure Compute Gallery - in this example we are using myGallery
sigName=myIbGallery
# name of the image definition to be created - in this example we are using myImageDef
imageDefName=myIbImageDef
# image distribution metadata reference name
runOutputName=aibLinuxSIG
```

Create a variable for your subscription ID.

```
subscriptionID=$(az account show --query id --output tsv)
```

Create the resource group.

```
az group create -n $sigResourceGroup -l $location
```

## Create a user-assigned identity and set permissions on the resource group

Image Builder will use the [user-identity](#) provided to inject the image into the Azure Compute Gallery. In this example, you will create an Azure role definition that has the granular actions to perform distributing the image to the gallery. The role definition will then be assigned to the user-identity.

```

# create user assigned identity for image builder to access the storage account where the script is located
identityName=aibBuiUserId$(date +'%s')
az identity create -g $sigResourceGroup -n $identityName

# get identity id
imgBuilderCliId=$(az identity show -g $sigResourceGroup -n $identityName --query clientId -o tsv)

# get the user identity URI, needed for the template
imgBuilderId=/subscriptions/$subscriptionID/resourcegroups/$sigResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/$identityName

# this command will download an Azure role definition template, and update the template with the parameters specified earlier.
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleImageCreation.json -o aibRoleImageCreation.json

imageRoleDefName="Azure Image Builder Image Def"$(date +'%s')

# update the definition
sed -i -e "s/<subscriptionID>/${subscriptionID}/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/${sigResourceGroup}/g" aibRoleImageCreation.json
sed -i -e "s/Azure Image Builder Service Image Creation Role/${imageRoleDefName}/g" aibRoleImageCreation.json

# create role definitions
az role definition create --role-definition ./aibRoleImageCreation.json

# grant role definition to the user assigned identity
az role assignment create \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup

```

## Create an image definition and gallery

To use Image Builder with an Azure Compute Gallery, you need to have an existing gallery and image definition. Image Builder will not create the gallery and image definition for you.

If you don't already have a gallery and image definition to use, start by creating them. First, create a gallery.

```

az sig create \
-g $sigResourceGroup \
--gallery-name $sigName

```

Then, create an image definition.

```

az sig image-definition create \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--publisher myIbPublisher \
--offer myOffer \
--sku 18.04-LTS \
--os-type Linux

```

## Download and configure the .json

Download the .json template and configure it with your variables.

```
curl  
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/1_Creating_a_Custom_Linux_Shared_Image_Gallery_Image/helloImageTemplateforSIG.json -o helloImageTemplateforSIG.json  
sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateforSIG.json  
sed -i -e "s/<rgName>/$sigResourceGroup/g" helloImageTemplateforSIG.json  
sed -i -e "s/<imageDefName>/$imageDefName/g" helloImageTemplateforSIG.json  
sed -i -e "s/<sharedImageGalName>/$sigName/g" helloImageTemplateforSIG.json  
sed -i -e "s/<region1>/$location/g" helloImageTemplateforSIG.json  
sed -i -e "s/<region2>/$additionalRegion/g" helloImageTemplateforSIG.json  
sed -i -e "s/<runOutputName>/$runOutputName/g" helloImageTemplateforSIG.json  
sed -i -e "s%<imgBuilderId>%$imgBuilderId%g" helloImageTemplateforSIG.json
```

## Create the image version

This next part will create the image version in the gallery.

Submit the image configuration to the Azure Image Builder service.

```
az resource create \  
--resource-group $sigResourceGroup \  
--properties @helloImageTemplateforSIG.json \  
--is-full-object \  
--resource-type Microsoft.VirtualMachineImages/imageTemplates \  
-n helloImageTemplateforSIG01
```

Start the image build.

```
az resource invoke-action \  
--resource-group $sigResourceGroup \  
--resource-type Microsoft.VirtualMachineImages/imageTemplates \  
-n helloImageTemplateforSIG01 \  
--action Run
```

Creating the image and replicating it to both regions can take a while. Wait until this part is finished before moving on to creating a VM.

## Create the VM

Create a VM from the image version that was created by Azure Image Builder.

```
az vm create \  
--resource-group $sigResourceGroup \  
--name myAibGalleryVM \  
--admin-username aibuser \  
--location $location \  
--image  
"/subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup/providers/Microsoft.Compute/galleries/$sigName/images/$imageDefName/versions/latest" \  
--generate-ssh-keys
```

SSH into the VM.

```
ssh aibuser@<publicIpAddress>
```

You should see the image was customized with a *Message of the Day* as soon as your SSH connection is established!

```
*****
** This VM was built from the:      **
** !! AZURE VM IMAGE BUILDER Custom Image !!      **
** You have just been Customized :-)      **
*****
```

## Clean up resources

If you want to now try re-customizing the image version to create a new version of the same image, skip the next steps and go on to [Use Azure Image Builder to create another image version](#).

This will delete the image that was created, along with all of the other resource files. Make sure you are finished with this deployment before deleting the resources.

When deleting gallery resources, you need delete all of the image versions before you can delete the image definition used to create them. To delete a gallery, you first need to have deleted all of the image definitions in the gallery.

Delete the image builder template.

```
az resource delete \
--resource-group $sigResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateforSIG01
```

Delete permissions assignments, roles and identity

```
az role assignment delete \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup

az role definition delete --name "$imageRoleDefName"

az identity delete --ids $imgBuilderId
```

Get the image version created by image builder, this always starts with `0.`, and then delete the image version

```
sigDefImgVersion=$(az sig image-version list \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID --query [].'name' -o json | grep 0. | tr -d ''')

az sig image-version delete \
-g $sigResourceGroup \
--gallery-image-version $sigDefImgVersion \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID
```

Delete the image definition.

```
az sig image-definition delete \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID
```

Delete the gallery.

```
az sig delete -r $sigName -g $sigResourceGroup
```

Delete the resource group.

```
az group delete -n $sigResourceGroup -y
```

## Next steps

Learn more about [Azure Compute Galleries](#).

# Create a Windows image and distribute it to an Azure Compute Gallery

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this article, you learn how to use Azure VM Image Builder and Azure PowerShell to create an image version in an [Azure Compute Gallery](#) (formerly Shared Image Gallery) and then distribute the image globally. You can also do this by using the [Azure CLI](#).

To configure the image, this article uses a JSON template, which you can find at [armTemplateWinSIG.json](#). You'll download and edit a local version of the template, so you'll also use a local PowerShell session.

To distribute the image to an Azure Compute Gallery, the template uses `sharedImage` as the value for the `distribute` section of the template.

VM Image Builder automatically runs `Sysprep` to generalize the image. The command is a generic `Sysprep` command, and you can [override](#) it if you need to.

Be aware of the number of times you layer customizations. You can run the `Sysprep` command a limited number of times on a single Windows image. After you've reached the `Sysprep` limit, you must re-create your Windows image. For more information, see [Limits on how many times you can run Sysprep](#).

## Register the features

To use VM Image Builder, you need to register the features.

1. Check your provider registrations. Make sure that each one returns *Registered*.

```
Get-AzResourceProvider -ProviderNamespace Microsoft.VirtualMachineImages | Format-table -Property ResourceTypes,RegistrationState
Get-AzResourceProvider -ProviderNamespace Microsoft.Storage | Format-table -Property ResourceTypes,RegistrationState
Get-AzResourceProvider -ProviderNamespace Microsoft.Compute | Format-table -Property ResourceTypes,RegistrationState
Get-AzResourceProvider -ProviderNamespace Microsoft.KeyVault | Format-table -Property ResourceTypes,RegistrationState
Get-AzResourceProvider -ProviderNamespace Microsoft.Network | Format-table -Property ResourceTypes,RegistrationState
```

2. If they don't return *Registered*, register the providers by running the following commands:

```
Register-AzResourceProvider -ProviderNamespace Microsoft.VirtualMachineImages
Register-AzResourceProvider -ProviderNamespace Microsoft.Storage
Register-AzResourceProvider -ProviderNamespace Microsoft.Compute
Register-AzResourceProvider -ProviderNamespace Microsoft.KeyVault
Register-AzResourceProvider -ProviderNamespace Microsoft.Network
```

3. Install PowerShell modules:

```
'Az.ImageBuilder', 'Az.ManagedServiceIdentity' | ForEach-Object {Install-Module -Name $_ -AllowPrerelease}
```

## Create variables

Because you'll be using some pieces of information repeatedly, create some variables to store that information.

Replace the values for the variables, such as `username` and `vmpassword`, with your own information.

```
# Get existing context
$currentAzContext = Get-AzContext

# Get your current subscription ID.
$subscriptionID=$currentAzContext.Subscription.Id

# Destination image resource group
$imageResourceGroup="aibwinsig"

# Location
$location="westus"

# Image distribution metadata reference name
$runOutputName="aibCustWinManImg02ro"

# Image template name
$imageTemplateName="helloImageTemplateWin02ps"

# Distribution properties object name (runOutput).
# This gives you the properties of the managed image on completion.
$runOutputName="winclientR01"

# Create a resource group for the VM Image Builder template and Azure Compute Gallery
New-AzResourceGroup ` 
    -Name $imageResourceGroup ` 
    -Location $location
```

## Create a user-assigned identity and set permissions on the resource group

VM Image Builder uses the provided [user-identity](#) to inject the image into Azure Compute Gallery. In this example, you create an Azure role definition with specific actions for distributing the image. The role definition is then assigned to the user identity.

```
# setup role def names, these need to be unique
$timeInt=$(get-date -UFormat "%s")
$imageRoleDefName="Azure Image Builder Image Def"+$timeInt
$identityName="aibIdentity"+$timeInt

## Add an Azure PowerShell module to support AzUserAssignedIdentity
Install-Module -Name Az.ManagedServiceIdentity

# Create an identity
New-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName

$identityNameResourceId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).Id
$identityNamePrincipalId=$(Get-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name
$identityName).PrincipalId
```

## Assign permissions for the identity to distribute the images

Use this command to download an Azure role definition template, and then update it with the previously specified parameters.

```

$aibRoleImageCreationUrl="https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Roles/aibRoleImageCreation.json"
$aibRoleImageCreationPath = "aibRoleImageCreation.json"

# Download the configuration
Invoke-WebRequest -Uri $aibRoleImageCreationUrl -OutFile $aibRoleImageCreationPath -UseBasicParsing

((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<subscriptionID>', $subscriptionID) | Set-Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace '<rgName>', $imageResourceGroup) | Set-Content -Path $aibRoleImageCreationPath
((Get-Content -path $aibRoleImageCreationPath -Raw) -replace 'Azure Image Builder Service Image Creation Role', $imageRoleDefName) | Set-Content -Path $aibRoleImageCreationPath

# Create a role definition
New-AzRoleDefinition -InputFile ./aibRoleImageCreation.json

# Grant the role definition to the VM Image Builder service principal
New-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $imageRoleDefName -Scope "/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"

```

#### NOTE

If you receive the error "New-AzRoleDefinition: Role definition limit exceeded. No more role definitions can be created," see [Troubleshoot Azure RBAC \(role-based access control\)](#).

## Create an Azure Compute Gallery

To use VM Image Builder with an Azure Compute Gallery, you need to have an existing gallery and image definition. VM Image Builder doesn't create the gallery and image definition for you.

If you don't already have a gallery and image definition to use, start by creating them.

```

# Gallery name
$sigGalleryName= "myIBSIG"

# Image definition name
$imageDefName ="winSvrimage"

# Additional replication region
$replRegion2="eastus"

# Create the gallery
New-AzGallery `

    -GalleryName $sigGalleryName `

    -ResourceGroupName $imageResourceGroup `

    -Location $location

# Create the image definition
New-AzGalleryImageDefinition `

    -GalleryName $sigGalleryName `

    -ResourceGroupName $imageResourceGroup `

    -Location $location `

    -Name $imageDefName `

    -OsState generalized `

    -OsType Windows `

    -Publisher 'myCompany' `

    -Offer 'WindowsServer' `

    -Sku 'WinSrv2019'

```

# Download and configure the template

Download the JSON template and configure it with your variables.

```
$templateFilePath = "armTemplateWinSIG.json"

Invoke-WebRequest `

    -Uri
    "https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/1_Creating_a_Custom_Win_Shared_Image_Gallery_Image/armTemplateWinSIG.json" `

    -OutFile $templateFilePath `

    -UseBasicParsing

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<subscriptionID>', $subscriptionID | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<rgName>', $imageResourceGroup | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<runOutputName>', $runOutputName | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<imageDefName>', $imageDefName | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<sharedImageGalName>', $sigGalleryName | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<region1>', $location | Set-Content -Path $templateFilePath

(Get-Content -path $templateFilePath -Raw ) `

    -replace '<region2>', $replRegion2 | Set-Content -Path $templateFilePath

((Get-Content -path $templateFilePath -Raw) -replace '<imgBuilderId>', $identityNameResourceId) | Set-Content -Path $templateFilePath
```

# Create the image version

Your template must be submitted to the service. The following commands will download any dependent artifacts, such as scripts, and store them in the staging resource group, which is prefixed with */T\_*.

```
New-AzResourceGroupDeployment `

    -ResourceGroupName $imageResourceGroup `

    -TemplateFile $templateFilePath `

    -ApiVersion "2020-02-14" `

    -imageTemplateName $imageTemplateName `

    -svclocation $location
```

To build the image, invoke 'Run' on the template.

```
Invoke-AzResourceAction `

    -ResourceName $imageTemplateName `

    -ResourceGroupName $imageResourceGroup `

    -ResourceType Microsoft.VirtualMachineImages/imageTemplates `

    -ApiVersion "2020-02-14" `

    -Action Run
```

Creating the image and replicating it to both regions can take a few moments. Before you begin creating a VM, wait until this part is finished.

```
Get-AzImageBuilderTemplate -ImageTemplateName $imageTemplateName -ResourceGroupName $imageResourceGroup | `

    Select-Object -Property Name, LastRunStatusRunState, LastRunStatusMessage, ProvisioningState
```

## Create the VM

Create a VM from the image version that you created with VM Image Builder.

1. Get the image version that you created:

```
$imageVersion = Get-AzGalleryImageVersion `  
-ResourceGroupName $imageResourceGroup `  
-GalleryName $sigGalleryName `  
-GalleryImageDefinitionName $imageDefName
```

2. Create the VM in the second region, where the image was replicated:

```
$vmResourceGroup = "myResourceGroup"  
$vmName = "myVMfromImage"  
  
# Create user object  
$cred = Get-Credential -Message "Enter a username and password for the virtual machine."  
  
# Create a resource group  
New-AzResourceGroup -Name $vmResourceGroup -Location $replRegion2  
  
# Network pieces  
$subnetConfig = New-AzVirtualNetworkSubnetConfig -Name mySubnet -AddressPrefix 192.168.1.0/24  
$vnet = New-AzVirtualNetwork -ResourceGroupName $vmResourceGroup -Location $replRegion2 `  
-Name MYvNET -AddressPrefix 192.168.0.0/16 -Subnet $subnetConfig  
$pip = New-AzPublicIpAddress -ResourceGroupName $vmResourceGroup -Location $replRegion2 `  
-Name "mypublicdns$(Get-Random)" -AllocationMethod Static -IdleTimeoutInMinutes 4  
$nsgRuleRDP = New-AzNetworkSecurityRuleConfig -Name myNetworkSecurityGroupRuleRDP -Protocol Tcp `  
-Direction Inbound -Priority 1000 -SourceAddressPrefix * -SourcePortRange * -DestinationAddressPrefix  
* `  
-DestinationPortRange 3389 -Access Deny  
$nsg = New-AzNetworkSecurityGroup -ResourceGroupName $vmResourceGroup -Location $replRegion2 `  
-Name myNetworkSecurityGroup -SecurityRules $nsgRuleRDP  
$nic = New-AzNetworkInterface -Name myNic -ResourceGroupName $vmResourceGroup -Location $replRegion2  
`  
-SubnetId $vnet.Subnets[0].Id -PublicIpAddressId $pip.Id -NetworkSecurityGroupId $nsg.Id  
  
# Create a virtual machine configuration using $imageVersion.Id to specify the image  
$vmConfig = New-AzVMConfig -VMName $vmName -VMSize Standard_D1_v2 | `  
Set-AzVMOperatingSystem -Windows -ComputerName $vmName -Credential $cred | `  
Set-AzVMSourceImage -Id $imageVersion.Id | `  
Add-AzVMNetworkInterface -Id $nic.Id  
  
# Create a virtual machine  
New-AzVM -ResourceGroupName $vmResourceGroup -Location $replRegion2 -VM $vmConfig
```

## Verify the customization

Create a Remote Desktop connection to the VM by using the username and password that you set when you created the VM. In the VM, open a Command Prompt window and run the following command:

```
dir c:\
```

You should see a directory named `buildActions` that was created during image customization.

## Clean up your resources

## NOTE

If you now want to try to recustomize the image version to create a new version of the same image, *skip the step outlined here* and go to [Use VM Image Builder to create another image version](#).

If you no longer need the resources that you created as you followed the process in this article, you can delete them.

The following process deletes both the image that you created and all the other resource files. Make sure that you've finished this deployment before you delete the resources.

Delete the resource group template first. Otherwise, the staging resource group (*/T\_*) that VM Image Builder uses won't be cleaned up.

1. Get the ResourceID of the image template.

```
$resTemplateId = Get-AzResource -ResourceName $imageTemplateName -ResourceGroupName  
$imageResourceGroup -ResourceType Microsoft.VirtualMachineImages/imageTemplates -ApiVersion "2020-02-  
14"
```

2. Delete image template.

```
Remove-AzResource -ResourceId $resTemplateId.ResourceId -Force
```

3. Delete the role assignment.

```
Remove-AzRoleAssignment -ObjectId $identityNamePrincipalId -RoleDefinitionName $imageRoleDefName -  
Scope "/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"
```

4. Remove the definitions.

```
Remove-AzRoleDefinition -Name "$identityNamePrincipalId" -Force -Scope  
"/subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup"
```

5. Delete the identity.

```
Remove-AzUserAssignedIdentity -ResourceGroupName $imageResourceGroup -Name $identityName -Force
```

6. Delete the resource group.

```
Remove-AzResourceGroup $imageResourceGroup -Force
```

## Next steps

To update the image version that you created in this article, see [Use VM Image Builder to create another image version](#).

# Create a new VM image from an existing image by using Azure VM Image Builder in Linux

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In this article, you learn how to update an existing image version in an [Azure Compute Gallery](#) (formerly Shared Image Gallery) and publish it to the gallery as a new image version.

To configure the image, you use a sample JSON template, [helloImageTemplateforSIGfromSIG.json](#).

## Register the features

To use VM Image Builder, you need to register the features.

1. Check your provider registrations. Make sure that each one returns *Registered*.

```
az provider show -n Microsoft.VirtualMachineImages | grep registrationState  
az provider show -n Microsoft.KeyVault | grep registrationState  
az provider show -n Microsoft.Compute | grep registrationState  
az provider show -n Microsoft.Storage | grep registrationState  
az provider show -n Microsoft.Network | grep registrationState
```

2. If they don't return *Registered*, register the providers by running the following commands:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Set variables and permissions

If you've already created an Azure Compute Gallery by using [Create an image and distribute it to an Azure Compute Gallery](#), you've already created some of the variables you need.

1. If you haven't already created the variables, run the following commands:

```
# Resource group name  
sigResourceGroup=ibLinuxGalleryRG  
# Gallery location  
location=westus2  
# Additional region to replicate the image version to  
additionalRegion=eastus  
# Name of the Azure Compute Gallery  
sigName=myIbGallery  
# Name of the image definition to use  
imageDefName=myIbImageDef  
# image distribution metadata reference name  
runOutputName=aibSIGLinuxUpdate
```

2. Create a variable for your subscription ID:

```
subscriptionID=$(az account show --query id --output tsv)
```

- Get the image version that you want to update:

```
sigDefImgVersionId=$(az sig image-version list \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID --query [].'id' -o tsv)
```

## Create a user-assigned identity and set permissions on the resource group

You've set up the user identity in an earlier example, so now you need to get the resource ID, which will be appended to the template.

```
#get identity used previously
imgBuilderId=$(az identity list -g $sigResourceGroup --query "[?contains(name, 'aibBuiUserId')].id" -o tsv)
```

If you already have an Azure Compute Gallery but didn't set it up by following an earlier example, you need to assign permissions for VM Image Builder to access the resource group so that it can access the gallery. For more information, see [Create an image and distribute it to an Azure Compute Gallery](#).

## Modify the helloImage example

You can review the JSON example you're about to use at [helloImageTemplateforSIGfromSIG.json](#). For information about the JSON file, see [Create an Azure VM Image Builder template](#).

- Download the JSON example, as shown in [Create a Linux image and distribute it to an Azure Compute Gallery by using the Azure CLI](#).
- Configure the JSON with your variables:

```
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/8_Creating_a_Custom_Linux_Shared_Image_Gallery_Image_from_SIG/helloImageTemplateforSIGfromSIG.json -o
helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<rgName>/$sigResourceGroup/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<imageDefName>/$imageDefName/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<sharedImageGalName>/$sigName/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s%<sigDefImgVersionId>%$sigDefImgVersionId%" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<region1>/$location/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<region2>/$additionalRegion/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s/<runOutputName>/$runOutputName/g" helloImageTemplateforSIGfromSIG.json
sed -i -e "s%<imgBuilderId>%$imgBuilderId%" helloImageTemplateforSIGfromSIG.json
```

## Create the image

- Submit the image configuration to the VM Image Builder service:

```
az resource create \
--resource-group $sigResourceGroup \
--properties @helloImageTemplateforSIGfromSIG.json \
--is-full-object \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateforSIGfromSIG01
```

## 2. Start the image build:

```
az resource invoke-action \
--resource-group $sigResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateforSIGfromSIG01 \
--action Run
```

Wait for the image to be built and replicated before you move along to the next step.

# Create the VM

## 1. Create the VM by doing the following:

```
az vm create \
--resource-group $sigResourceGroup \
--name aibImgVm001 \
--admin-username azureuser \
--location $location \
--image
"/subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup/providers/Microsoft.Compute/galleries/$sigName/images/$imageDefName/versions/latest" \
--generate-ssh-keys
```

## 2. Create a Secure Shell (SSH) connection to the VM by using the public IP address of the VM.

```
ssh azureuser@<pubIp>
```

After the SSH connection is established, you should receive a "Message of the Day" saying that the image was customized:

```
*****
**      This VM was built from the:      **
**      !! AZURE VM IMAGE BUILDER Custom Image !!      **
**      You have just been Customized :-)      **
*****
```

## 3. Type `exit` to close the SSH connection.

## 4. To list the image versions that are now available in your gallery, run:

```
az sig image-version list -g $sigResourceGroup -r $sigName -i $imageDefName -o table
```

# Next steps

To learn more about the components of the JSON file that you used in this article, see [Create an Azure VM Image Builder template](#).

# Create a new Windows VM image from an existing image by using Azure VM Image Builder

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this article, you learn how to update an existing Windows image version in an [Azure Compute Gallery](#) (formerly Shared Image Gallery) and publish it to the gallery as a new image version.

To configure the image, you use a sample JSON template, [helloImageTemplateforSIGfromWinSIG.json](#).

## Register the features

To use VM Image Builder, you need to register the features.

1. Check your provider registrations. Make sure that each one returns *Registered*.

```
az provider show -n Microsoft.VirtualMachineImages | grep registrationState  
az provider show -n Microsoft.KeyVault | grep registrationState  
az provider show -n Microsoft.Compute | grep registrationState  
az provider show -n Microsoft.Storage | grep registrationState  
az provider show -n Microsoft.Network | grep registrationState
```

2. If they don't return *Registered*, register the providers by running the following commands:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Set variables and permissions

If you've already created an Azure Compute Gallery by using [Create an image and distribute it to an Azure Compute Gallery](#), you've already created some of the variables you need.

### NOTE

VM Image Builder supports creating custom images only in the same resource group that the source-managed image is in. In the following example, update the resource group name, *ibsigRG*, with the name of resource group that your source-managed image is in.

1. If you haven't already created the variables, run the following commands:

```
# Resource group name - we are using ibsigRG in this example
sigResourceGroup=myIBWinRG
# Datacenter location - we are using West US 2 in this example
location=westus
# Additional region to replicate the image to - we are using East US in this example
additionalRegion=eastus
# name of the Azure Compute Gallery - in this example we are using myGallery
sigName=my22stsSIG
# name of the image definition to be created - in this example we are using myImageDef
imageDefName=winSvrImages
# image distribution metadata reference name
runOutputName=w2019SigRo
# User name and password for the VM
username="user name for the VM"
vmpassword="password for the VM"
```

## 2. Create a variable for your subscription ID:

```
subscriptionID=$(az account show --query id --output tsv)
```

## 3. Get the image version that you want to update:

```
sigDefImgVersionId=$(az sig image-version list \
-g $sigResourceGroup \
--gallery-name $sigName \
--gallery-image-definition $imageDefName \
--subscription $subscriptionID --query [].'id' -o tsv)
```

# Create a user-assigned identity and set permissions on the resource group

You've set up the user identity in an earlier example, so now you need to get the resource ID, which will be appended to the template.

```
#get identity used previously
imgBuilderId=$(az identity list -g $sigResourceGroup --query "[?contains(name, 'aibBuiUserId')].id" -o tsv)
```

If you already have an Azure Compute Gallery but didn't set it up by following an earlier example, you need to assign permissions for VM Image Builder to access the resource group so that it can access the gallery. For more information, see [Create an image and distribute it to an Azure Compute Gallery](#).

## Modify the helolimage example

You can review the JSON example you're about to use at [helolimageTemplateforSIGfromSIG.json](#). For information about the JSON file, see [Create an Azure VM Image Builder template](#).

1. Download the JSON example, as shown in [Create a user-assigned identity and set permissions on the resource group](#).
2. Configure the JSON with your variables:

```
curl  
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/8_Creating_a_Custom_  
Win_Shared_Image_Gallery_Image_from_SIG/helloImageTemplateforSIGfromWinSIG.json -o  
helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<rgName>/$sigResourceGroup/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<imageDefName>/$imageDefName/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<sharedImageGalName>/$signInName/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s%<sigDefImgVersionId>%$sigDefImgVersionId%g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<region1>/$location/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<region2>/$additionalRegion/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s/<runOutputName>/$runOutputName/g" helloImageTemplateforSIGfromWinSIG.json  
sed -i -e "s%<imgBuilderId>%$imgBuilderId%g" helloImageTemplateforSIGfromWinSIG.json
```

## Create the image

1. Submit the image configuration to the VM Image Builder service:

```
az resource create \  
    --resource-group $sigResourceGroup \  
    --location $location \  
    --properties @helloImageTemplateforSIGfromWinSIG.json \  
    --is-full-object \  
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \  
    -n imageTemplateforSIGfromWinSIG01
```

2. Start the image build:

```
az resource invoke-action \  
    --resource-group $sigResourceGroup \  
    --resource-type Microsoft.VirtualMachineImages/imageTemplates \  
    -n imageTemplateforSIGfromWinSIG01 \  
    --action Run
```

Wait for the image to be built and replicated before you move along to the next step.

## Create the VM

Create the VM by doing the following:

```
az vm create \  
    --resource-group $sigResourceGroup \  
    --name aibImgWinVm002 \  
    --admin-username $username \  
    --admin-password $vmpassword \  
    --image  
    "/subscriptions/$subscriptionID/resourceGroups/$sigResourceGroup/providers/Microsoft.Compute/galleries/$sigN  
ame/images/$imageDefName/versions/latest" \  
    --location $location
```

## Verify the customization

Create a Remote Desktop connection to the VM by using the username and password you set when you created the VM. Inside the VM, open a Command Prompt window, and then run:

```
dir c:\
```

You should now see two directories:

- *buildActions*: Created in the first image version.
- *buildActions2*: Created when you updated the first image version to create the second image version.

## Next steps

To learn more about the components of the JSON file that you used in this article, see [Create an Azure VM Image Builder template](#).

# Create an image and use a user-assigned managed identity to access files in an Azure storage account

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows how to create a customized image by using Azure VM Image Builder. The service uses a [user-assigned managed identity](#) to access files in an Azure storage account, without your having to make the files publicly accessible.

Azure VM Image Builder supports using scripts and copying files from GitHub, Azure storage accounts, and other locations. If you want to use the locations, they must be externally accessible to VM Image Builder.

In the following example, you'll create two resource groups, one for the custom image and the other to host an Azure storage account that contains a script file. This example simulates a real-life scenario, where you might have build artifacts or image files in various storage accounts. You'll create a user-assigned identity and then grant the identity read permissions on the script file, but you won't allow public access to the file. You'll then use the shell customizer to download and run a script from the storage account.

## Register the features

1. To use VM Image Builder, you need to register the feature:

```
az feature register --namespace Microsoft.VirtualMachineImages --name VirtualMachineTemplatePreview
```

2. Check the status of the feature registration:

```
az feature show --namespace Microsoft.VirtualMachineImages --name VirtualMachineTemplatePreview | grep state
```

3. Check your registration:

```
az provider show -n Microsoft.VirtualMachineImages | grep registrationState  
az provider show -n Microsoft.KeyVault | grep registrationState  
az provider show -n Microsoft.Compute | grep registrationState  
az provider show -n Microsoft.Storage | grep registrationState  
az provider show -n Microsoft.Network | grep registrationState
```

4. If the output doesn't show your features as *Registered*, run the following commands:

```
az provider register -n Microsoft.VirtualMachineImages  
az provider register -n Microsoft.Compute  
az provider register -n Microsoft.KeyVault  
az provider register -n Microsoft.Storage  
az provider register -n Microsoft.Network
```

## Create a resource group

1. Because you'll be using some pieces of information repeatedly, create some variables to store that

information.

```
# Image resource group name
imageResourceGroup=aibmdimsi
# Storage resource group
strResourceGroup=aibmdimsistor
# Location
location=WestUS2
# Name of the image to be created
imageName=aibCustLinuxImgMsi01
# Image distribution metadata reference name
runOutputName=u1804ManImgMsiro
```

2. Create a variable for your subscription ID:

```
subscriptionID=$(az account show --query id --output tsv)
```

3. Create resource groups for both the image and the script storage:

```
# Create a resource group for the image template
az group create -n $imageResourceGroup -l $location
# Create a resource group for the script storage
az group create -n $strResourceGroup -l $location
```

4. Create a user-assigned identity, and set permissions on the resource group:

VM Image Builder uses the provided [user identity](#) to inject the image into the resource group. In this example, you create an Azure role definition with specific actions for distributing the image. The role definition is then assigned to the user identity.

```
# Create a user-assigned identity for VM Image Builder to access the storage account where the script
is located
identityName=aibBuiUserId$(date +'%s')
az identity create -g $imageResourceGroup -n $identityName

# Get an identity ID
imgBuilderCliId=$(az identity show -g $imageResourceGroup -n $identityName --query clientId -o tsv)

# Get the user-identity URI, which is needed for the template
imgBuilderId=/subscriptions/$subscriptionID/resourcegroups/$imageResourceGroup/providers/Microsoft.Ma
nagedIdentity/userAssignedIdentities/$identityName

# Download the preconfigured role definition example
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/solutions/12_Creating_AIB_Security_Ro
les/aibRoleImageCreation.json -o aibRoleImageCreation.json

# Update the definition
sed -i -e "s/<subscriptionID>/$subscriptionID/g" aibRoleImageCreation.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" aibRoleImageCreation.json

# Create role definitions
az role definition create --role-definition ./aibRoleImageCreation.json

# Grant the role definition to the user-assigned identity
az role assignment create \
--assignee $imgBuilderCliId \
--role "Azure Image Builder Service Image Creation Role" \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup
```

5. Create the storage account, and copy the sample script into it from GitHub:

```
# Script storage account
scriptStorageAcc=aibstorscript$(date +'%s')

# Script container
scriptStorageAccContainer=scriptscont$(date +'%s')

# Script URL
scriptUrl=https://$scriptStorageAcc.blob.core.windows.net/$scriptStorageAccContainer/customizeScript.sh

# Create the storage account and blob in the resource group
az storage account create -n $scriptStorageAcc -g $strResourceGroup -l $location --sku Standard_LRS

az storage container create -n $scriptStorageAccContainer --fail-on-exist --account-name $scriptStorageAcc

# Copy in an example script from the GitHub repo
az storage blob copy start \
    --destination-blob customizeScript.sh \
    --destination-container $scriptStorageAccContainer \
    --account-name $scriptStorageAcc \
    --source-uri
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/customizeScript.sh
```

6. Give VM Image Builder permission to create resources in the image resource group. The `--assignee` value is the user-identity ID.

```
az role assignment create \
    --assignee $imgBuilderCliId \
    --role "Storage Blob Data Reader" \
    --scope
/subscriptions/$subscriptionID/resourceGroups/$strResourceGroup/providers/Microsoft.Storage/storageAccounts/$scriptStorageAcc/blobServices/default/containers/$scriptStorageAccContainer
```

## Modify the example

Download the example JSON file and configure it with the variables you created earlier.

```
curl
https://raw.githubusercontent.com/azure/azvmimagebuilder/master/quickstarts/7_Creating_Custom_Image_using_MSIs_to_Access_Storage/helloImageTemplateMsi.json -o helloImageTemplateMsi.json
sed -i -e "s/<subscriptionID>/$subscriptionID/g" helloImageTemplateMsi.json
sed -i -e "s/<rgName>/$imageResourceGroup/g" helloImageTemplateMsi.json
sed -i -e "s/<region>/$location/g" helloImageTemplateMsi.json
sed -i -e "s/<imageName>/$imageName/g" helloImageTemplateMsi.json
sed -i -e "s%<scriptUrl>%$scriptUrl%g" helloImageTemplateMsi.json
sed -i -e "s%<imgBuilderId>%$imgBuilderId%g" helloImageTemplateMsi.json
sed -i -e "s%<runOutputName>%$runOutputName%g" helloImageTemplateMsi.json
```

## Create the image

1. Submit the image configuration to the VM Image Builder service:

```
az resource create \
--resource-group $imageResourceGroup \
--properties @helloImageTemplateMsi.json \
--is-full-object \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateMsi01
```

## 2. Start the image build:

```
az resource invoke-action \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateMsi01 \
--action Run
```

The build can take about 15 minutes to finish.

## Create a VM

### 1. Create a VM from the image:

```
az vm create \
--resource-group $imageResourceGroup \
--name aibImgVm00 \
--admin-username aibuser \
--image $imageName \
--location $location \
--generate-ssh-keys
```

### 2. After the VM has been created, start a Secure Shell (SSH) session with it.

```
ssh aibuser@<publicIp>
```

After the SSH connection is established, you should receive a "Message of the Day" saying that the image was customized:

```
*****
**          This VM was built from the:      **
**      !! AZURE VM IMAGE BUILDER Custom Image !!  **
**          You have just been Customized :-)      **
*****
```

## Clean up your resources

If you no longer need the resources that were created during this process, you can delete them by running the following code:

```
az role definition delete --name "$imageRoleDefName"
```azurecli-interactive
az role assignment delete \
--assignee $imgBuilderCliId \
--role "$imageRoleDefName" \
--scope /subscriptions/$subscriptionID/resourceGroups/$imageResourceGroup
az identity delete --ids $imgBuilderId
az resource delete \
--resource-group $imageResourceGroup \
--resource-type Microsoft.VirtualMachineImages/imageTemplates \
-n helloImageTemplateMsi01
az group delete -n $imageResourceGroup
az group delete -n $strResourceGroup
```

## Next steps

If you have any problems using VM Image Builder, see [Troubleshoot Azure VM Image Builder](#).

# What's new in Azure VM Image Builder

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article contains all major API changes and feature updates for the Azure VM Image Builder service.

## API releases

### Version 2022-02-14

#### Improvements

- [Validation support](#)
  - Shell (Linux): Script or inline
  - PowerShell (Windows): Script or inline, run elevated, run as system
  - Source-Validation-Only mode
- [Customized staging resource group support](#)

### Version 2021-10-01

#### Breaking change

API version 2021-10-01 introduces a change to the error schema that will be part of every future API release. If you have any Azure VM Image Builder automations, be aware of the [new error output](#) when you switch to API version 2021-10-01 or later. We recommend, after you've switched to the latest API version, that you don't revert to an earlier version, because you'll have to change your automation again to produce the earlier error schema. We don't anticipate that we'll change the error schema again in future releases.

Error output for version 2020-02-14 and earlier

```
{  
  "code": "ValidationFailed",  
  "message": "Validation failed: 'ImageTemplate.properties.source': Field 'imageId' has a bad value:  
  '/subscriptions/subscriptionID/resourceGroups/resourceGroupName/providers/Microsoft.Compute/images/imageName  
  '. Please review http://aka.ms/azvmimagebuildertmplref for details on fields requirements in the Image  
  Builder Template."  
}
```

Error output for version 2021-10-01 and later

```
{  
  "error": {  
    "code": "ValidationFailed",  
    "message": "Validation failed: 'ImageTemplate.properties.source': Field 'imageId' has a bad value:  
    '/subscriptions/subscriptionID/resourceGroups/resourceGroupName/providers/Microsoft.Compute/images/imageName  
    '. Please review http://aka.ms/azvmimagebuildertmplref for details on fields requirements in the Image  
    Builder Template."  
  }  
}
```

#### Improvements

- Added support for [Build VM MSIs](#).
- Added support for Proxy VM size customization.

**Version 2020-02-14**

## Improvements

- Added support for creating images from the following sources:
  - Managed image
  - Azure Compute Gallery
  - Platform Image Repository (including Platform Image Purchase Plan)
- Added support for the following customizations:
  - Shell (Linux): Script or inline
  - PowerShell (Windows): Script or inline, run elevated, run as system
  - File (Linux and Windows)
  - Windows Restart (Windows)
  - Windows Update (Windows): Search criteria, filters, and update limit
- Added support for the following distribution types:
  - VHD (virtual hard disk)
  - Managed image
  - Azure Compute Gallery
- Other features:
  - Added support for customers to use their own virtual network
  - Added support for customers to customize the build VM (VM size, operating system disk size)
  - Added support for user-assigned Microsoft Windows Installer (MSI) (for customize/distribute steps)
  - Added support for [Gen2 images](#)

## Preview APIs

The following APIs are deprecated, but still supported:

- Version 2019-05-01-preview

## Next steps

Learn more about [VM Image Builder](#).

# Troubleshoot Azure VM Image Builder

9/21/2022 • 27 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Use this article to troubleshoot and resolve common issues that you might encounter when you're using Azure VM Image Builder.

## Prerequisites

When you're creating a build, do the following:

- The VM Image Builder service communicates to the build VM by using WinRM or Secure Shell (SSH). Do *not* disable these settings as part of the build.
- VM Image Builder creates resources as part of the build. Be sure to verify that Azure Policy doesn't prevent VM Image Builder from creating or using necessary resources.
  - Create an IT\_ resource group.
  - Create a storage account without a firewall.
- Verify that Azure Policy doesn't install unintended features on the build VM, such as Azure Extensions.
- Ensure that VM Image Builder has the correct permissions to read/write images and to connect to the storage account. For more information, review the permissions documentation for the [Azure CLI](#) or [Azure PowerShell](#).
- VM Image Builder will fail the build if the scripts or inline commands fail with errors (non-zero exit codes). Ensure that you've tested the custom scripts and verified that they run without error (exit code 0) or require user input. For more information, see [Create an Azure Virtual Desktop image by using VM Image Builder and PowerShell](#).

VM Image Builder failures can happen in two areas:

- During image template submission
- During image building

## Troubleshoot image template submission errors

Image template submission errors are returned at submission only. There isn't an error log for image template submission errors. If there's an error during submission, you can return the error by checking the status of the template, specifically by reviewing `ProvisioningState` and `ProvisioningErrorMessage` / `provisioningError`.

```
az image builder show --name $imageTemplateName --resource-group $imageResourceGroup
```

```
Get-AzImageBuilderTemplate -ImageTemplateName <imageTemplateName> -ResourceGroupName <imageTemplateResourceGroup> | Select-Object ProvisioningState, ProvisioningErrorMessage
```

### NOTE

For PowerShell, you'll need to install the [VM Image Builder PowerShell modules](#).

## IMPORTANT

API version 2021-10-01 introduces a change to the error schema that will be part of every future API release. If you have any Azure VM Image Builder automations, be aware of the [new error output](#) when you switch to API version 2021-10-01 or later. We recommend, after you've switched to the latest API version, that you don't revert to an earlier version, because you'll have to change your automation again to produce the earlier error schema. We don't anticipate that we'll change the error schema again in future releases.

Error output for version 2020-02-14 and earlier

```
{  
    "code": "ValidationFailed",  
    "message": "Validation failed: 'ImageTemplate.properties.source': Field 'imageId' has a bad value:  
'/subscriptions/subscriptionID/resourceGroups/resourceGroupName/providers/Microsoft.Compute/images/imageName  
. Please review http://aka.ms/azvmimagebuildertmplref for details on fields requirements in the Image  
Builder Template."  
}
```

Error output for version 2021-10-01 and later

```
{  
    "error": {  
        "code": "ValidationFailed",  
        "message": "Validation failed: 'ImageTemplate.properties.source': Field 'imageId' has a bad value:  
'/subscriptions/subscriptionID/resourceGroups/resourceGroupName/providers/Microsoft.Compute/images/imageName  
. Please review http://aka.ms/azvmimagebuildertmplref for details on fields requirements in the Image  
Builder Template."  
    }  
}
```

The following sections present problem resolution guidance for common image template submission errors.

## Update or upgrade of image templates is currently not supported

### Error

```
'Conflict'. Details: Update/Upgrade of image templates is currently not supported
```

### Cause

The template already exists.

### Solution

If you submit an image configuration template and the submission fails, a failed template artifact still exists. Delete the failed template.

## The resource operation finished with a terminal provisioning state of "Failed"

### Error

```
Microsoft.VirtualMachineImages/imageTemplates 'helloImageTemplateforSIG01' failed with message '{  
    "status": "Failed",  
    "error": {  
        "code": "ResourceDeploymentFailure",  
        "message": "The resource operation completed with terminal provisioning state 'Failed'.",  
        "details": [  
            {  
                "code": "InternalOperationError",  
                "message": "Internal error occurred."  
            }  
        ]  
    }  
}
```

### Cause

In most cases, the resource deployment failure error occurs because of missing permissions. This error may also

be caused by a conflict with the staging resource group.

#### Solution

Depending on your scenario, VM Image Builder might need permissions to:

- The source image or Azure Compute Gallery (formerly Shared Image Gallery) resource group.
- The distribution image or Azure Compute Gallery resource.
- The storage account, container, or blob that the `File` customizer is accessing.

Also, ensure the staging resource group name is uniquely specified for each image template.

For more information about configuring permissions, see [Configure VM Image Builder permissions by using the Azure CLI](#) or [Configure VM Image Builder permissions by using PowerShell](#).

## Error getting a managed image

#### Error

```
Build (Managed Image) step failed: Error getting Managed Image
'/subscriptions/.../providers/Microsoft.Compute/images/mymanagedmg1': Error getting managed image (...):
compute.
ImagesClient#Get: Failure responding to request: StatusCode=403 -- Original Error: autorest/azure: Service
returned an error.
Status=403 Code="AuthorizationFailed" Message="The client '.....' with object id '.....' doesn't have
authorization to perform action 'Microsoft.Compute/images/read' over scope
```

#### Cause

Missing permissions.

#### Solution

Depending on your scenario, VM Image Builder might need permissions to:

- The source image or Azure Compute Gallery resource group.
- The distribution image or Azure Compute Gallery resource.
- The storage account, container, or blob that the `File` customizer is accessing.

For more information about configuring permissions, see [Configure VM Image Builder permissions by using the Azure CLI](#) or [Configure VM Image Builder permissions by using PowerShell](#).

## The build step failed for the image version

#### Error

```
Build (Shared Image Version) step failed for Image Version
'/subscriptions/.../providers/Microsoft.Compute/galleries/.../images/... /versions/0.23768.4001': Error
getting Image Version
'/subscriptions/.../resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/.../images/.../versions/0.
23768.4001': Error getting image version '... :0.23768.4001': compute.GalleryImageVersionsClient#Get:
Failure responding to request: StatusCode=404 -- Original Error: autorest/azure: Service returned an error.
Status=404 Code="ResourceNotFound" Message="The Resource
'Microsoft.Compute/galleries/.../images/.../versions/0.23768.4001' under resource group '<rgName>' was not
found."
```

#### Cause

VM Image Builder can't locate the source image.

#### Solution

Ensure that the source image is correct and exists in the location of VM Image Builder.

## Downloading an external file to a local file

#### Error

```
Downloading external file (<myFile>) to local file (xxxxx.0.customizer.fp) [attempt 1 of 10] failed: Error  
downloading '<myFile>' to 'xxxxx.0.customizer.fp'..
```

## Cause

The file name or location is incorrect, or the location isn't reachable.

## Solution

Ensure that the file is reachable. Verify that the name and location are correct.

## Authorization error creating disk

The Azure Image Builder build fails with an authorization error that looks like the following:

### Error

```
Attempting to deploy created Image template in Azure fails with an 'The client '6df325020-fe22-4e39-bd69-  
10873965ac04' with object id '6df325020-fe22-4e39-bd69-10873965ac04' does not have authorization to perform  
action 'Microsoft.Compute/disks/write' over scope  
'/subscriptions/<subscriptionID>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/disks/proxyV  
mDiskWin_<timestampl>' or the scope is invalid. If access was recently granted, please refresh your  
credentials.'
```

## Cause

This error is caused when trying to specify a pre-existing resource group and VNet to the Azure Image Builder service with a Windows source image.

## Solution

You will need to assign the contributor role to the resource group for the service principal corresponding to Azure Image Builder's first party app by using the CLI command or portal instructions below.

First, validate that the service principal is associated with Azure Image Builder's first party app by using the following CLI command:

```
az ad sp show --id {servicePrincipalName, or objectId}
```

Then, to implement this solution using CLI, use the following command:

```
az role assignment create -g {ResourceGroupName} --assignee {AibrpSpOid} --role Contributor
```

To implement this solution in portal, follow the instructions in this documentation: [Assign Azure roles using the Azure portal - Azure RBAC](#).

For [Step 1: Identify the needed scope](#): The needed scope is your resource group.

For [Step 3: Select the appropriate role](#): The role is Contributor.

For [Step 4: Select who needs access](#): Select member "Azure Virtual Machine Image Builder"

Then proceed to [Step 6: Assign role](#) to assign the role.

## Troubleshoot build failures

For image build failures, get the error from the `lastrunstatus`, and then review the details in the `customization.log` file.

```
az image builder show --name $imageTemplateName --resource-group $imageResourceGroup
```

```
Get-AzImageBuilderTemplate -ImageTemplateName <imageTemplateName> -ResourceGroupName  
<imageTemplateResourceGroup> | Select-Object LastRunStatus, LastRunStatusMessage
```

## Customization log

When the image build is running, logs are created and stored in a storage account. VM Image Builder creates the storage account in the temporary resource group when you create an image template artifact.

The storage account name uses the pattern `IT_<ImageResourceGroupName><TemplateName><GUID>` (for example, `IT_aibmdi_helloImageTemplateLinux01`).

To view the `customization.log` file in the resource group, select **Storage Account > Blobs > [packerlogs]**, select **directory**, and then select the `customization.log` file.

## Understand the customization log

The log is verbose. It covers the image build, including any issues with the image distribution, such as Azure Compute Gallery replication. These errors are surfaced in the error message of the image template status.

The `customization.log` file includes the following stages:

1. *Deploy the build VM and dependencies by using ARM templates to the IT\_staging resource group stage.*

This stage includes multiple POSTs to the VM Image Builder resource provider:

```
Azure request method="POST"  
request="https://management.azure.com/subscriptions/<subID>/resourceGroups/IT_aibImageRG200_window201  
9VnetTemplate01_dec33089-1cc3-cccc-cccc-ccccccc/providers/Microsoft.Storage/storageAccounts  
..  
PACKER OUT ==> azure-arm: Deploying deployment template ...  
..
```

2. *Status of the deployments stage.* This stage includes the status of each resource deployment:

```
PACKER ERR 2020/04/30 23:28:50 packer: 2020/04/30 23:28:50 Azure request method="GET"  
request="https://management.azure.com/subscriptions/<subID>/resourcegroups/IT_aibImageRG200_window201  
9VnetTemplate01_dec33089-1cc3-4505-ae28-  
6661e43fac48/providers/Microsoft.Resources/deployments/pkrdp51lc0339jg/operationStatuses/085861331762  
07523519?[REDACTED]" body=""
```

3. *Connect to the build VM stage.*

In Windows, VM Image Builder connects by using WinRM:

```
PACKER ERR 2020/04/30 23:30:50 packer: 2020/04/30 23:30:50 Waiting for WinRM, up to timeout: 10m0s  
..  
PACKER OUT      azure-arm: WinRM connected.
```

In Linux, VM Image Builder connects by using SSH:

```
PACKER OUT ==> azure-arm: Waiting for SSH to become available...  
PACKER ERR 2019/12/10 17:20:51 packer: 2020/04/10 17:20:51 [INFO] Waiting for SSH, up to timeout:  
20m0s  
PACKER OUT ==> azure-arm: Connected to SSH!
```

4. *Run customizations stage.* When customizations run, you can identify them by reviewing the `customization.log` file. Search for (`telemetry`).

```
(telemetry) Starting provisioner windows-update
(telemetry) ending windows-update
(telemetry) Starting provisioner powershell
(telemetry) ending powershell
(telemetry) Starting provisioner file
(telemetry) ending file
(telemetry) Starting provisioner windows-restart
(telemetry) ending windows-restart

(telemetry) Finalizing. - This means the build hasfinished
```

5. *Deprovision* stage. VM Image Builder adds a hidden customizer. This deprovision step is responsible for preparing the VM for deprovisioning. In Windows, it runs `Sysprep` (by using `c:\DeprovisioningScript.ps1`). In Linux, it runs `waagent-deprovision` (by using `/tmp/DeprovisioningScript.sh`).

For example:

```
PACKER ERR 2020/03/04 23:05:04 [INFO] (telemetry) Starting provisioner powershell
PACKER ERR 2020/03/04 23:05:04 packer: 2020/03/04 23:05:04 Found command: if( TEST-PATH
c:\DeprovisioningScript.ps1 ){cat c:\DeprovisioningScript.ps1} else {echo "Deprovisioning script
[c:\DeprovisioningScript.ps1] could not be found. Image build may fail or the VM created from the
Image may not boot. Please make sure the deprovisioning script is not accidentally deleted by a
Customizer in the Template."}
```

6. *Cleanup* stage. After the build has finished, the VM Image Builder resources are deleted.

```
PACKER ERR ==> azure-arm: Deleting individual resources ...
...
PACKER ERR 2020/02/04 02:04:23 packer: 2020/02/04 02:04:23 Azure request method="DELETE"
request="https://management.azure.com/subscriptions/<subId>/resourceGroups/IT_aibDevOpsImg_t_vvvvvv_
yyyyyy-de5f-4f7c-92f2-xxxxxxxx/providers/Microsoft.Network/networkInterfaces/pkrnijamvpo08eo?
[REDACTED]" body=""
...
PACKER ERR ==> azure-arm: The resource group was not created by Packer, not deleting ...
```

## Tips for troubleshooting script or inline customization

- Test the code before you supply it to VM Image Builder.
- Ensure that Azure Policy and Firewall allow connectivity to remote resources.
- Output comments to the console by using `Write-Host` or `echo`. Doing so lets you search the `customization.log` file.

## Troubleshoot common build errors

### Packer build command failure

#### Error

```
"provisioningState": "Succeeded",
"lastRunStatus": {
  "startTime": "2020-04-30T23:24:06.756985789Z",
  "endTime": "2020-04-30T23:39:14.268729811Z",
  "runState": "Failed",
  "message": "Failed while waiting for packerizer: Microservice has failed: Failed while processing request: Error when executing packerizer: Packer build command has failed: exit status 1. During the image build, a failure has occurred, please review the build log to identify which build/customization step failed. For more troubleshooting steps go to https://aka.ms/azvmimagebuilderts. Image Build log location: https://xxxxxxxxx.blob.core.windows.net/packerlogs/xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxx/customization.log. OperationId: xxxxxx-5a8c-4379-xxxx-8d85493bc791. Use this operationId to search packer logs."
```

## Cause

Customization failure.

## Solution

Review the log to locate customizer failures. Search for *(telemetry)*.

For example:

```
(telemetry) Starting provisioner windows-update
(telemetry) ending windows-update
(telemetry) Starting provisioner powershell
(telemetry) ending powershell
(telemetry) Starting provisioner file
(telemetry) ending file
(telemetry) Starting provisioner windows-restart
(telemetry) ending windows-restart

(telemetry) Finalizing. - This means the build has finished
```

## Time-out exceeded

### Error

```
Deployment failed. Correlation ID: xxxxx-xxxx-xxxx-xxxx-xxxxxxxxx. Failed in building/customizing image: Failed while waiting for packerizer: Timeout waiting for microservice to complete: 'context deadline exceeded'
```

## Cause

The build exceeded the build time-out. This error is seen in the 'lastrunstatus'.

## Solution

1. Review the *customization.log* file. Identify the last customizer to run. Search for *(telemetry)*, starting from the bottom of the log.
2. Check script customizations. The customizations might not be suppressing user interaction for commands, such as `quiet` options. For example, `apt-get install -y` results in the script execution waiting for user interaction.
3. If you're using the `File` customizer to download artifacts greater than 20 MB, see workarounds section.
4. Review errors and dependencies in script that might cause the script to wait.
5. If you expect that the customizations need more time, increase the value of `buildTimeoutInMinutes`. The default is 4 hours.
6. If you have resource-intensive actions, such as downloading gigabytes (GB) of files, consider the underlying build VM size. The service uses a Standard\_D1\_v2 VM. The VM has 1 vCPU and 3.5 GB of memory. If you're downloading 50 GB, you'll likely exhaust the VM resources and cause communication

failures between VM Image Builder and the build VM. Retry the build with a larger-memory VM by setting the [VM\\_size](#).

## Long file download time

### Error

```
[086cf9c4-0457-4e8f-bfd4-908cf3fe43c] PACKER OUT  
myBigFile.zip 826 B / 826000 B 1.00%  
[086cf9c4-0457-4e8f-bfd4-908cf3fe43c] PACKER OUT  
myBigFile.zip 1652 B / 826000 B 2.00%  
[086cf9c4-0457-4e8f-bfd4-908cf3fe43c] PACKER OUT  
..  
hours later...  
..  
myBigFile.zip 826000 B / 826000 B 100.00%  
[086cf9c4-0457-4e8f-bfd4-908cf3fe43c] PACKER OUT
```

### Cause

[File](#) customizer is downloading a large file.

### Solution

[File](#) customizer is suitable only for small (less than 20 MB) file downloads. For larger file downloads, use a script or inline command. For example, in Linux you can use [wget](#) or [curl](#). In Windows, you can use [Invoke-WebRequest](#).

## Error waiting on Azure Compute Gallery

### Error

```
Deployment failed. Correlation ID: XXXXXX-XXXX-XXXXXX-XXXX-XXXXXX. Failed in distributing 1 images out of total 1: {[Error 0] [Distribute 0] Error publishing MDI to Azure Compute Gallery:/subscriptions/<subId>/resourceGroups/xxxxxx/providers/Microsoft.Compute/galleries/xxxxx/images/xxxx xx, Location:eastus. Error: Error returned from SIG client while publishing MDI to Azure Compute Gallery for dstImageLocation: eastus, dstSubscription: <subId>, dstResourceGroupName: XXXXXX, dstGalleryName: XXXXXX, dstGalleryImageName: XXXXXX. Error: Error waiting on Azure Compute Gallery future for resource group: XXXXXX, gallery name: XXXXXX, gallery image name: XXXXXX. Error: Future#WaitForCompletion: context has been cancelled: StatusCode=200 -- Original Error: context deadline exceeded}
```

### Cause

VM Image Builder timed out waiting for the image to be added and replicated to Azure Compute Gallery. If the image is being injected into the gallery, you can assume that the image build was successful. However, the overall process failed because VM Image Builder was waiting on Azure Compute Gallery to complete the replication. Even though the build has failed, the replication continues. You can get the properties of the image version by checking the distribution *runOutput*.

```
$runOutputName=<distributionRunOutput>  
az resource show \  
    --ids  
    "/subscriptions/$subscriptionID/resourcegroups/$imageResourceGroup/providers/Microsoft.VirtualMachineImages/  
    imageTemplates/$imageTemplateName/runOutputs/$runOutputName" \  
    --api-version=2020-02-14
```

### Solution

Increase the value of [buildTimeoutInMinutes](#).

## Low Windows resource information events

### Error

```
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT      azure-arm: Waiting for operation to complete (system  
performance: 1% cpu; 37% memory)... .
```

```
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT    azure-arm: Waiting for operation to complete (system  
performance: 51% cpu; 35% memory)...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT    azure-arm: Waiting for operation to complete (system  
performance: 21% cpu; 37% memory)...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT    azure-arm: Waiting for operation to complete (system  
performance: 21% cpu; 36% memory)...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT    azure-arm: Waiting for operation to complete (system  
performance: 90% cpu; 32% memory)...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT    azure-arm: Waiting for the Windows Modules Installer  
to exit...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer: 2020/04/30 23:38:58 [INFO]  
command 'PowerShell -ExecutionPolicy Bypass -OutputFormat Text -File C:/Windows/Temp/packer-windows-update-  
elevated.ps1' exited with code: 101  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT ==> azure-arm: Restarting the machine...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer: 2020/04/30 23:38:58 [INFO] RPC  
endpoint: Communicator ended with: 101  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] 1672 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] 0 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] RPC client: Communicator ended  
with: 101  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] RPC endpoint: Communicator  
ended with: 101  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT ==> azure-arm: Waiting for machine to become available...  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] 1672 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] 0 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] RPC client: Communicator ended with: 101  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer: 2020/04/30 23:38:58 [INFO]  
starting remote command: shutdown.exe -f -r -t 0 -c "packer restart"  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer: 2020/04/30 23:38:58 [INFO]  
command 'shutdown.exe -f -r -t 0 -c "packer restart"' exited with code: 0  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer: 2020/04/30 23:38:58 [INFO] RPC  
endpoint: Communicator ended with: 0  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] 0 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] 0 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT ==> azure-arm: A system shutdown is in progress.(1115)  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] RPC client: Communicator ended  
with: 0  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 [INFO] RPC endpoint: Communicator  
ended with: 0  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] 0 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] 0 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:58 packer-provisioner-windows-update:  
2020/04/30 23:38:58 [INFO] RPC client: Communicator ended with: 0  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer: 2020/04/30 23:38:59 [INFO]  
starting remote command: shutdown.exe -f -r -t 60 -c "packer restart test"  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer: 2020/04/30 23:38:59 [INFO]  
command 'shutdown.exe -f -r -t 60 -c "packer restart test"' exited with code: 1115  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer: 2020/04/30 23:38:59 [INFO] RPC  
endpoint: Communicator ended with: 1115  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 [INFO] 0 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 [INFO] 40 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 [INFO] RPC client: Communicator ended  
with: 1115  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 [INFO] RPC endpoint: Communicator  
ended with: 1115  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer-provisioner-windows-update:  
2020/04/30 23:38:59 [INFO] 40 bytes written for 'stderr'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer-provisioner-windows-update:  
2020/04/30 23:38:59 [INFO] 0 bytes written for 'stdout'  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer-provisioner-windows-update:  
2020/04/30 23:38:59 [INFO] RPC client: Communicator ended with: 1115  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER ERR 2020/04/30 23:38:59 packer-provisioner-windows-update:  
2020/04/30 23:38:59 Retryable error: Machine not yet available (exit status 1115)  
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT Build 'azure-arm' errored: unexpected EOF
```

```
[45f485cf-5a8c-4379-9937-8d85493bc791] PACKER OUT
```

## Cause

Resource exhaustion. This issue is commonly seen with Windows Update running with the default build VM size D1\_V2.

## Solution

Increase the build VM size.

## The build finished but no artifacts were created

### Error

```
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT Build 'azure-arm' errored: Future#WaitForCompletion: context has been cancelled: StatusCode=200 -- Original Error: context deadline exceeded
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR ==> Some builds didn't complete successfully and had errors:
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR 2020/04/30 22:29:23 machine readable: azure-arm,error
[]string{"Future#WaitForCompletion: context has been cancelled: StatusCode=200 -- Original Error: context deadline exceeded"}
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT ==> Some builds didn't complete successfully and had errors:
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR ==> Builds finished but no artifacts were created.
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT --> azure-arm: Future#WaitForCompletion: context has been cancelled: StatusCode=200 -- Original Error: context deadline exceeded
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR 2020/04/30 22:29:23 Cancelling builder after context cancellation context canceled
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR 2020/04/30 22:29:23 [INFO] (telemetry) Finalizing.
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT ==> Builds finished but no artifacts were created.
[a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER ERR 2020/04/30 22:29:24 waiting for all plugin processes to complete...
Done exporting Packer logs to Azure for Packer prefix: [a170b40d-2d77-4ac3-8719-72cdc35cf889] PACKER OUT
```

## Cause

The build timed out while it was waiting for the required Azure resources to be created.

## Solution

Rerun the build to try again.

## Resource not found

### Error

```
"provisioningState": "Succeeded",
"lastRunStatus": {
  "startTime": "2020-05-01T00:13:52.599326198Z",
  "endTime": "2020-05-01T00:15:13.62366898Z",
  "runState": "Failed",
  "message": "network.InterfacesClient#UpdateTags: Failure responding to request: StatusCode=404 -- Original Error: autorest/azure: Service returned an error. Status=404 Code=\"ResourceNotFound\" Message=\"The Resource 'Microsoft.Network/networkInterfaces/aibpls7lz2e.nic.4609d697-be0a-4cb0-86af-49b6fe877fe1' under resource group 'IT_aibImageRG200_window2019VnetTemplate01_9988723b-af56-413a-9006-84130af0e9df' was not found.\\""
},
```

## Cause

Missing permissions.

## Solution

Recheck to ensure that VM Image Builder has all the permissions it requires.

For more information about configuring permissions, see [Configure VM Image Builder permissions by using the](#)

Azure CLI or Configure VM Image Builder permissions by using PowerShell.

## Sysprep timing

Error

```
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER OUT      azurite-arm. IMAGE_STATE_UNDEPLOYABLE
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER ERR 2020/05/05 22:26:17 Cancelling builder after context cancellation context canceled
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER OUT Cancelling build after receiving terminated
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER ERR 2020/05/05 22:26:17 packer: 2020/05/05 22:26:17 Cancelling provisioning due to context cancellation: context canceled
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER OUT ==> azure-arm:
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER ERR 2020/05/05 22:26:17 packer: 2020/05/05 22:26:17 Cancelling hook after context cancellation context canceled
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER OUT ==> azure-arm: The resource group was not created by Packer, deleting individual resources ...
[922bdf36-b53c-4e78-9cd8-6b70b9674685] PACKER ERR ==> azure-arm: The resource group was not created by Packer, deleting individual resources ...
```

## Cause

The cause might be a timing issue because of the D1\_V2 VM size. If customizations are limited and are run in less than three seconds, `Sysprep` commands are run by VM Image Builder to deprovision. When VM Image Builder deprovisions, the `Sysprep` command checks for the *WindowsAzureGuestAgent*, which might not be fully installed and might cause the timing issue.

## Solution

To avoid the timing issue, you can increase the VM size or you can add a 60-second PowerShell sleep customization.

## The build is canceled after the context cancellation context is canceled

### Error

```
PACKER ERR 2020/03/26 22:11:23 Cancelling builder after context cancellation context canceled
PACKER OUT Cancelling build after receiving terminated
PACKER ERR 2020/03/26 22:11:23 packer-builder-azure-arm plugin: Cancelling hook after context cancellation context canceled
..
PACKER ERR 2020/03/26 22:11:23 packer-builder-azure-arm plugin: Cancelling provisioning due to context cancellation: context canceled
PACKER ERR 2020/03/26 22:11:25 packer-builder-azure-arm plugin: [ERROR] Remote command exited without exit status or exit signal.
PACKER ERR 2020/03/26 22:11:25 packer-builder-azure-arm plugin: [INFO] RPC endpoint: Communicator ended with: 2300218
PACKER ERR 2020/03/26 22:11:25 [INFO] 148974 bytes written for 'stdout'
PACKER ERR 2020/03/26 22:11:25 [INFO] 0 bytes written for 'stderr'
PACKER ERR 2020/03/26 22:11:25 [INFO] RPC client: Communicator ended with: 2300218
PACKER ERR 2020/03/26 22:11:25 [INFO] RPC endpoint: Communicator ended with: 2300218
```

## Cause

VM Image Builder uses port 22 (Linux) or 5986 (Windows) to connect to the build VM. This occurs when the service is disconnected from the build VM during an image build. The reasons for the disconnection can vary, but enabling or configuring a firewall in the script can block the previously mentioned ports.

## Solution

Review your scripts for firewall changes or enablement, or changes to SSH or WinRM, and ensure that any changes allow for constant connectivity between the service and the build VM on the previously mentioned ports. For more information, see [VM Image Builder networking options](#).

## JWT errors in log early in the build

### Error

Early in the build process, the build fails and the log indicates a JSON Web Token (JWT) error:

```
PACKER OUT Error: Failed to prepare build: "azure-arm"
PACKER ERR
PACKER OUT
PACKER ERR * client_jwt will expire within 5 minutes, please use a JWT that is valid for at least 5 minutes
PACKER OUT 1 error(s) occurred:
```

## Cause

The `buildTimeoutInMinutes` value in the template is set to from 1 to 5 minutes.

## Solution

As described in [Create an VM Image Builder template](#), the time-out must be set to 0 to use the default or set to more than 5 minutes to override the default. Change the time-out in your template to 0 to use the default or to a minimum of 6 minutes.

## Resource deletion errors

### Error

Intermediate resources are cleaned up toward the end of the build, and the customization log might show several resource deletion errors:

```
PACKER OUT ==> azure-arm: Error deleting resource. Will retry.
...
PACKER OUT ==> azure-arm: Error: network.PublicIPAddressesClient#Delete: Failure sending request:
StatusCode=0 -- Original Error: Code="PublicIPAddressCannotBeDeleted" Message=...
...
PACKER ERR 2022/03/07 18:43:06 packer-plugin-azure plugin: 2022/03/07 18:43:06 Retryable error:
network.SecurityGroupsClient#Delete: Failure sending request: StatusCode=0 -- Original Error:
Code="InUseNetworkSecurityGroupCannotBeDeleted"...
```

## Cause

These error log messages are mostly harmless, because resource deletions are retried several times and, ordinarily, they eventually succeed. You can verify this by continuing to follow the deletion logs until you observe a success message. Alternatively, you can inspect the staging resource group to confirm whether the resource has been deleted.

Making these observations is especially important in build failures, where these error messages might lead you to conclude that they're the reason for the failures, even when the actual errors might be elsewhere.

## DevOps tasks

### Troubleshoot the task

The task fails only if an error occurs during customization. When this happens, the task reports the failure and leaves the staging resource group, with the logs, so that you can identify the issue.

To locate the log, you need to know the template name. Go to **pipeline > failed build**, and then drill down into the VM Image Builder DevOps task.

You'll see the log and a template name:

```
start reading task parameters...
found build at: /home/vsts/work/r1/a/_ImageBuilding/webapp
end reading parameters
getting storage account details for aibstordot1556933914
created archive /home/vsts/work/_temp/temp_web_package_21475337782320203.zip
Source for image: { type: 'SharedImageVersion',
  imageVersionId:
  '/subscriptions/<subscriptionID>/resourceGroups/<rgName>/providers/Microsoft.Compute/galleries/<galleryName>/images/<imageDefName>/versions/<imgVersionNumber>' }
template name: t_1556938436xxx
```

1. Go to the Azure portal, search for the template name in the resource group, and then search for the resource group by typing **IT\_\***.
2. Select the storage account name > **blobs** > **containers** > **logs**.

### Troubleshoot successful builds

You might occasionally need to investigate successful builds and review their logs. As mentioned earlier, if the image build is successful, the staging resource group that contains the logs will be deleted as part of the cleanup. To prevent an automatic cleanup, though, you can introduce a `sleep` after the inline command, and then view the logs as the build is paused. To do so, do the following:

1. Update the inline command by adding **Write-Host / Echo "Sleep"**. This gives you time to search in the log.
2. Add a `sleep` value of at least 10 minutes by using a **Start-Sleep** or `sleep` Linux command.
3. Use this method to identify the log location, and then keep downloading or checking the log until it gets to `sleep`.

### Operation was canceled

#### Error

```
2020-05-05T18:28:24.9280196Z ##[section]Starting: Azure VM Image Builder Task
2020-05-05T18:28:24.9609966Z =====
2020-05-05T18:28:24.9610739Z Task : Azure VM Image Builder Test
2020-05-05T18:28:24.9611277Z Description : Build images using Azure Image Builder resource provider.
2020-05-05T18:28:24.9611608Z Version : 1.0.18
2020-05-05T18:28:24.9612003Z Author : Microsoft Corporation
2020-05-05T18:28:24.9612718Z Help : For documentation, and end to end example, please visit:
http://aka.ms/azvmmagebuilderdevops
2020-05-05T18:28:24.9613390Z =====
2020-05-05T18:28:26.0651512Z start reading task parameters...
2020-05-05T18:28:26.0673377Z found build at: d:\a\r1\appsAndImageBuilder\webApp
2020-05-05T18:28:26.0708785Z end reading parameters
2020-05-05T18:28:26.0745447Z getting storage account details for aibstagstor1565047758
2020-05-05T18:28:29.8812270Z created archive d:\a\_temp\temp_web_package_09737279437949953.zip
2020-05-05T18:28:33.1568013Z Source for image: { type: 'PlatformImage',
2020-05-05T18:28:33.1584131Z publisher: 'MicrosoftWindowsServer',
2020-05-05T18:28:33.1585965Z offer: 'WindowsServer',
2020-05-05T18:28:33.1592768Z sku: '2016-Datacenter',
2020-05-05T18:28:33.1594191Z version: '14393.3630.2004101604' }
2020-05-05T18:28:33.1595387Z template name: t_1588703313152
2020-05-05T18:28:33.1597453Z starting put template...
2020-05-05T18:28:52.9278603Z put template: Succeeded
2020-05-05T18:28:52.9281282Z starting run template...
2020-05-05T19:33:14.3923479Z ##[error]The operation was canceled.
2020-05-05T19:33:14.3939721Z ##[section]Finishing: Azure VM Image Builder Task
```

#### Cause

If the build wasn't canceled by a user, it was canceled by Azure DevOps User Agent. Most likely, the 1-hour time-out has occurred because of Azure DevOps capabilities. If you're using a private project and agent, you get 60 minutes of build time. If the build exceeds the time-out, DevOps cancels the running task.

For more information about Azure DevOps capabilities and limitations, see [Microsoft-hosted agents](#).

### Solution

You can host your own DevOps agents or look to reduce the time of your build. For example, if you're distributing to Azure Compute Gallery, you can replicate them to one region or replicate them asynchronously.

## Slow Windows logon

### Error

This error might occur when you create a Windows 10 image by using VM Image Builder, create a VM from the image, and then use Remote Desktop Protocol (RDP). You wait several minutes at the first logon screen, and then a blue screen displays the following message:



Please wait for the Windows Modules Installer

### Solution

1. In the image build, check to ensure that:
  - There are no outstanding reboots required by adding a Windows Restart customizer as the last customization.
  - All software installation is complete.
2. Add the `/mode:vm` option to the default `Sysprep` that VM Image Builder uses. For more information, go to the "Override the commands" section under "VMs created from VM Image Builder images aren't created successfully."

## VMs created from VM Image Builder images aren't created successfully

By default, VM Image Builder runs *deprovision* code at the end of each image customization phase to *generalize* the image. To generalize an image is to set it up to reuse to create multiple VMs. As part of the process, you can pass in VM settings, such as hostname, username, and so on. In Windows, VM Image Builder runs `Sysprep`, and in Linux, VM Image Builder runs `waagent -deprovision`.

In Windows, VM Image Builder uses a generic `Sysprep` command. However, this command might not be suitable for every successful Windows generalization. With VM Image Builder, you can customize the `Sysprep` command. Note that VM Image Builder is an image automation tool that's responsible for running `Sysprep` command successfully. But you might need different `Sysprep` commands to make your image reusable. In Linux, VM Image Builder uses a generic `waagent -deprovision+user` command. For more information, see [Microsoft Azure Linux Agent documentation](#).

If you're migrating an existing customization and you're using various `Sysprep` or `waagent` commands, you can try the VM Image Builder generic commands. If the VM creation fails, use your previous `Sysprep` or `waagent` commands.

Let's suppose you've used VM Image Builder successfully to create a Windows custom image, but you've failed to create a VM successfully from the image. For example, the VM creation fails to finish or it times out. In this event, do either of the following:

- Review the Windows Server Sysprep documentation.
- Raise a support request with the Windows Server Sysprep Customer Services Support team. They can help troubleshoot your issue and advise you on the correct `Sysprep` command.

### Command locations and file names

In Windows:

```
c:\DeprovisioningScript.ps1
```

In Linux:

```
/tmp/DeprovisioningScript.sh
```

### The `Sysprep` command: Windows

```
Write-Output '">>>> Waiting for GA Service (RdAgent) to start ...'
while ((Get-Service RdAgent).Status -ne 'Running') { Start-Sleep -s 5 }
Write-Output '">>>> Waiting for GA Service (WindowsAzureTelemetryService) to start ...'
while ((Get-Service WindowsAzureTelemetryService) -and ((Get-Service WindowsAzureTelemetryService).Status -ne 'Running')) { Start-Sleep -s 5 }
Write-Output '">>>> Waiting for GA Service (WindowsAzureGuestAgent) to start ...'
while ((Get-Service WindowsAzureGuestAgent).Status -ne 'Running') { Start-Sleep -s 5 }
Write-Output '">>>> Sysprepping VM ...'
if( Test-Path $Env:SystemRoot\system32\Sysprep\unattend.xml ) {
    Remove-Item $Env:SystemRoot\system32\Sysprep\unattend.xml -Force
}
& $Env:SystemRoot\System32\Sysprep\Sysprep.exe /oobe /generalize /quiet /quit
while($true) {
    $imageState = (Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Setup\State).ImageState
    Write-Output $imageState
    if ($imageState -eq 'IMAGE_STATE_GENERALIZE_RESEAL_TO_OOBE') { break }
    Start-Sleep -s 5
}
Write-Output '">>>> Sysprep complete ...'
```

### The `-deprovision` command: Linux

```
/usr/sbin/waagent -force -deprovision+user && export HISTSIZE=0 && sync
```

## Override the commands

To override the commands, use the PowerShell or shell script provisoners to create the command files with the exact file name and put them in the previously listed directories. VM Image Builder reads these commands and writes output to the *customization.log* file.

## Get support

If you've referred to the guidance and are still having problems, you can open a support case. Be sure to select the correct product and support topic. Doing so will ensure that you're connected with the Azure VM Image Builder support team.

Selecting the case product:

```
Product Family: Azure
Product: Virtual Machine Running (Window\Linux)
Support Topic: Azure Features
Support Subtopic: Azure Image Builder
```

## Next steps

For more information, see [VM Image Builder overview](#).

# How to use Packer to create Linux virtual machine images in Azure

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Each virtual machine (VM) in Azure is created from an image that defines the Linux distribution and OS version. Images can include pre-installed applications and configurations. The Azure Marketplace provides many first and third-party images for most common distributions and application environments, or you can create your own custom images tailored to your needs. This article details how to use the open source tool [Packer](#) to define and build custom images in Azure.

## NOTE

Azure now has a service, Azure Image Builder, for defining and creating your own custom images. Azure Image Builder is built on Packer, so you can even use your existing Packer shell provisioner scripts with it. To get started with Azure Image Builder, see [Create a Linux VM with Azure Image Builder](#).

## Create Azure resource group

During the build process, Packer creates temporary Azure resources as it builds the source VM. To capture that source VM for use as an image, you must define a resource group. The output from the Packer build process is stored in this resource group.

Create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create -n myResourceGroup -l eastus
```

## Create Azure credentials

Packer authenticates with Azure using a service principal. An Azure service principal is a security identity that you can use with apps, services, and automation tools like Packer. You control and define the permissions as to what operations the service principal can perform in Azure.

Create a service principal with [az ad sp create-for-rbac](#) and output the credentials that Packer needs:

```
az ad sp create-for-rbac --role Contributor --scopes /subscriptions/<subscription_id> --query "{ client_id: appId, client_secret: password, tenant_id: tenant }"
```

An example of the output from the preceding commands is as follows:

```
{
  "client_id": "f5b6a5cf-fbdf-4a9f-b3b8-3c2cd00225a4",
  "client_secret": "0e760437-bf34-4aad-9f8d-870be799c55d",
  "tenant_id": "72f988bf-86f1-41af-91ab-2d7cd011db47"
}
```

To authenticate to Azure, you also need to obtain your Azure subscription ID with `az account show`:

```
az account show --query "{ subscription_id: id }"
```

You use the output from these two commands in the next step.

## Define Packer template

To build images, you create a template as a JSON file. In the template, you define builders and provisioners that carry out the actual build process. Packer has a [provisioner for Azure](#) that allows you to define Azure resources, such as the service principal credentials created in the preceding step.

Create a file named `ubuntu.json` and paste the following content. Enter your own values for the following parameters:

PARAMETER	WHERE TO OBTAIN
<code>client_id</code>	First line of output from <code>az ad sp create</code> command - <code>appId</code>
<code>client_secret</code>	Second line of output from <code>az ad sp create</code> command - <code>password</code>
<code>tenant_id</code>	Third line of output from <code>az ad sp create</code> command - <code>tenant</code>
<code>subscription_id</code>	Output from <code>az account show</code> command
<code>managed_image_resource_group_name</code>	Name of resource group you created in the first step
<code>managed_image_name</code>	Name for the managed disk image that is created

```
{
  "builders": [
    {
      "type": "azure-arm",
      "client_id": "f5b6a5cf-fbdf-4a9f-b3b8-3c2cd00225a4",
      "client_secret": "0e760437-bf34-4aad-9f8d-870be799c55d",
      "tenant_id": "72f988bf-86f1-41af-91ab-2d7cd011db47",
      "subscription_id": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
      "managed_image_resource_group_name": "myResourceGroup",
      "managed_image_name": "myPackerImage",
      "os_type": "Linux",
      "image_publisher": "Canonical",
      "image_offer": "UbuntuServer",
      "image_sku": "16.04-LTS",
      "azure_tags": {
        "dept": "Engineering",
        "task": "Image deployment"
      },
      "location": "East US",
      "vm_size": "Standard_DS2_v2"
    ],
    "provisioners": [
      {
        "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E sh '{{ .Path }}'",  

        "inline": [
          "apt-get update",
          "apt-get upgrade -y",
          "apt-get -y install nginx",
          "/usr/sbin/waagent -force -deprovision+user && export HISTSIZE=0 && sync"
        ],
        "inline_shebang": "/bin/sh -x",
        "type": "shell"
      }
    ]
}
```

You can also create a file named *ubuntu.pkr.hcl* and paste the following content with your own values as used for the above parameters table.

```

source "azure-arm" "autogenerated_1" {
  azure_tags = {
    dept = "Engineering"
    task = "Image deployment"
  }
  client_id           = "f5b6a5cf-fbdf-4a9f-b3b8-3c2cd00225a4"
  client_secret        = "0e760437-bf34-4aad-9f8d-870be799c55d"
  image_offer          = "UbuntuServer"
  image_publisher       = "Canonical"
  image_sku             = "16.04-LTS"
  location              = "East US"
  managed_image_name     = "myPackerImage"
  managed_image_resource_group_name = "myResourceGroup"
  os_type                = "Linux"
  subscription_id        = "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
  tenant_id              = "72f988bf-86f1-41af-91ab-2d7cd011db47"
  vm_size                = "Standard_DS2_v2"
}

build {
  sources = ["source.azure-arm.autogenerated_1"]

  provisioner "shell" {
    execute_command = "chmod +x {{ .Path }}; {{ .Vars }} sudo -E sh '{{ .Path }}'"
    inline         = ["apt-get update", "apt-get upgrade -y", "apt-get -y install nginx",
      "/usr/sbin/waagent -force -deprovision+user && export HISTSIZE=0 && sync"]
    inline_shbang   = "/bin/sh -x"
  }
}

```

This template builds an Ubuntu 16.04 LTS image, installs NGINX, then deprovisions the VM.

#### NOTE

If you expand on this template to provision user credentials, adjust the provisioner command that deprovisions the Azure agent to read `-deprovision` rather than `deprovision+user`. The `+user` flag removes all user accounts from the source VM.

## Build Packer image

If you don't already have Packer installed on your local machine, [follow the Packer installation instructions](#).

Build the image by specifying your Packer template file as follows:

```
./packer build ubuntu.json
```

You can also build the image by specifying the `ubuntu.pkr.hcl` file as follows:

```
packer build ubuntu.pkr.hcl
```

An example of the output from the preceding commands is as follows:

```

azure-arm output will be in this color.

==> azure-arm: Running builder ...
    azure-arm: Creating Azure Resource Manager (ARM) client ...
==> azure-arm: Creating resource group ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> Location        : 'East US'
==> azure-arm: -> Tags          :
==> azure-arm: ->> dept : Engineering
==> azure-arm: ->> task : Image deployment
==> azure-arm: Validating deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> DeploymentName   : 'pkrdpswtxmqm7ly'
==> azure-arm: Deploying deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> DeploymentName   : 'pkrdpswtxmqm7ly'
==> azure-arm: Getting the VM's IP address ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> PublicIPAddressName : 'packerPublicIP'
==> azure-arm: -> NicName         : 'packerNic'
==> azure-arm: -> Network Connection : 'PublicEndpoint'
==> azure-arm: -> IP Address       : '40.76.218.147'
==> azure-arm: Waiting for SSH to become available...
==> azure-arm: Connected to SSH!
==> azure-arm: Provisioning with shell script: /var/folders/h1/ymh5bdx15wgdn5hvgj1wc0zh0000gn/T/packer-
shell1868574263
    azure-arm: WARNING! The waagent service will be stopped.
    azure-arm: WARNING! Cached DHCP leases will be deleted.
    azure-arm: WARNING! root password will be disabled. You will not be able to login as root.
    azure-arm: WARNING! /etc/resolvconf/resolv.conf.d/tail and /etc/resolvconf/resolv.conf.d/original will
be deleted.
    azure-arm: WARNING! packer account and entire home directory will be deleted.
==> azure-arm: Querying the machine's properties ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> ComputeName      : 'pkrvmswtxmqm7ly'
==> azure-arm: -> Managed OS Disk  : '/subscriptions/guid/resourceGroups/packer-Resource-Group-
swtxmqm7ly/providers/Microsoft.Compute/disks/osdisk'
==> azure-arm: Powering off machine ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> ComputeName      : 'pkrvmswtxmqm7ly'
==> azure-arm: Capturing image ...
==> azure-arm: -> Compute ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: -> Compute Name       : 'pkrvmswtxmqm7ly'
==> azure-arm: -> Compute Location     : 'East US'
==> azure-arm: -> Image ResourceGroupName : 'myResourceGroup'
==> azure-arm: -> Image Name         : 'myPackerImage'
==> azure-arm: -> Image Location       : 'eastus'
==> azure-arm: Deleting resource group ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-swtxmqm7ly'
==> azure-arm: Deleting the temporary OS disk ...
==> azure-arm: -> OS Disk : skipping, managed disk was used...
Build 'azure-arm' finished.

==> Builds finished. The artifacts of successful builds are:
--> azure-arm: Azure.ResourceManagement.VMImage:

ManagedImageResourceGroupName: myResourceGroup
ManagedImageName: myPackerImage
ManagedImageLocation: eastus

```

It takes a few minutes for Packer to build the VM, run the provisioners, and clean up the deployment.

## Create VM from Azure Image

You can now create a VM from your Image with [az vm create](#). Specify the Image you created with the `--image`

parameter. The following example creates a VM named *myVM* from *myPackerImage* and generates SSH keys if they don't already exist:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image myPackerImage \
--admin-username azureuser \
--generate-ssh-keys
```

If you wish to create VMs in a different resource group or region than your Packer image, specify the image ID rather than image name. You can obtain the image ID with [az image show](#).

It takes a few minutes to create the VM. Once the VM has been created, take note of the `publicIpAddress` displayed by the Azure CLI. This address is used to access the NGINX site via a web browser.

To allow web traffic to reach your VM, open port 80 from the Internet with [az vm open-port](#):

```
az vm open-port \
--resource-group myResourceGroup \
--name myVM \
--port 80
```

## Test VM and NGINX

Now you can open a web browser and enter `http://publicIpAddress` in the address bar. Provide your own public IP address from the VM create process. The default NGINX page is displayed as in the following example:



## Next steps

You can also use existing Packer provisioner scripts with [Azure Image Builder](#).

# PowerShell: How to use Packer to create virtual machine images in Azure

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Each virtual machine (VM) in Azure is created from an image that defines the Windows distribution and OS version. Images can include pre-installed applications and configurations. The Azure Marketplace provides many first and third-party images for most common OS' and application environments, or you can create your own custom images tailored to your needs. This article details how to use the open-source tool [Packer](#) to define and build custom images in Azure.

This article was last tested on 8/5/2020 using [Packer](#) version 1.6.1.

## NOTE

Azure now has a service, Azure Image Builder, for defining and creating your own custom images. Azure Image Builder is built on Packer, so you can even use your existing Packer shell provisioner scripts with it. To get started with Azure Image Builder, see [Create a Windows VM with Azure Image Builder](#).

## Create Azure resource group

During the build process, Packer creates temporary Azure resources as it builds the source VM. To capture that source VM for use as an image, you must define a resource group. The output from the Packer build process is stored in this resource group.

Create a resource group with [New-AzResourceGroup](#). The following example creates a resource group named *myPackerGroup* in the *eastus* location:

```
$rgName = "myPackerGroup"  
$location = "East US"  
New-AzResourceGroup -Name $rgName -Location $location
```

## Create Azure credentials

Packer authenticates with Azure using a service principal. An Azure service principal is a security identity that you can use with apps, services, and automation tools like Packer. You control and define the permissions as to what operations the service principal can perform in Azure.

Create a service principal with [New-AzADServicePrincipal](#). The value for `-DisplayName` needs to be unique; replace with your own value as needed.

```
$sp = New-AzADServicePrincipal -DisplayName "PackerPrincipal" -role Contributor -scope  
/subscriptions/yyyyyyyy-yyyy-yyyy-yyyy-yyyyyyyy  
$plainPassword = (New-AzADSPrinciple -ObjectId $sp.Id).SecretText
```

Then output the password and application ID.

```
$plainPassword  
$sp.AppId
```

To authenticate to Azure, you also need to obtain your Azure tenant and subscription IDs with [Get-AzSubscription](#):

```
$subName = "mySubscriptionName"  
$sub = Get-AzSubscription -SubscriptionName $subName
```

## Define Packer template

To build images, you create a template as a JSON file. In the template, you define builders and provisioners that carry out the actual build process. Packer has a [builder for Azure](#) that allows you to define Azure resources, such as the service principal credentials created in the preceding step.

Create a file named *windows.json* and paste the following content. Enter your own values for the following:

PARAMETER	WHERE TO OBTAIN
<i>client_id</i>	View service principal ID with <code>\$sp.AppId</code>
<i>client_secret</i>	View the auto-generated password with <code>\$plainPassword</code>
<i>tenant_id</i>	Output from <code>\$sub.TenantId</code> command
<i>subscription_id</i>	Output from <code>\$sub.SubscriptionId</code> command
<i>managed_image_resource_group_name</i>	Name of resource group you created in the first step
<i>managed_image_name</i>	Name for the managed disk image that is created

```
{
  "builders": [
    {
      "type": "azure-arm",
      "client_id": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
      "client_secret": "ppppppp-pppp-pppp-pppppppppp",
      "tenant_id": "zzzzzz-zzzz-zzzz-zzzzzzzzzz",
      "subscription_id": "yyyyyy-yyyy-yyyy-yyyyyyyyyy",
      "managed_image_resource_group_name": "myPackerGroup",
      "managed_image_name": "myPackerImage",
      "os_type": "Windows",
      "image_publisher": "MicrosoftWindowsServer",
      "image_offer": "WindowsServer",
      "image_sku": "2016-Datacenter",
      "communicator": "winrm",
      "winrm_use_ssl": true,
      "winrm_insecure": true,
      "winrm_timeout": "5m",
      "winrm_username": "packer",
      "azure_tags": {
        "dept": "Engineering",
        "task": "Image deployment"
      },
      "build_resource_group_name": "myPackerGroup",
      "vm_size": "Standard_D2_v2"
    ],
    "provisioners": [
      {
        "type": "powershell",
        "inline": [
          "Add-WindowsFeature Web-Server",
          "while ((Get-Service RdAgent).Status -ne 'Running') { Start-Sleep -s 5 }",
          "while ((Get-Service WindowsAzureGuestAgent).Status -ne 'Running') { Start-Sleep -s 5 }",
          "& $env:SystemRoot\\System32\\Sysprep\\Sysprep.exe /oobe /generalize /quiet /quit",
          "while($true) { $imageState = Get-ItemProperty HKLM:\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Setup\\State | Select ImageState;
          if($imageState.ImageState -ne 'IMAGE_STATE_GENERALIZE_RESEAL_TO_OOBE') { Write-Output $imageState.ImageState; Start-Sleep -s 10 } else { break } }"
        ]
      }
    ]
  }
}
```

You can also create a file named *windows.pkr.hcl* and paste the following content with your own values as used for the above parameters table.

```

source "azure-arm" "autogenerated_1" {
  azure_tags = {
    dept = "Engineering"
    task = "Image deployment"
  }
  build_resource_group_name      = "myPackerGroup"
  client_id                     = "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
  client_secret                 = "ppppppp-pppp-pppp-pppp-ppppppppppp"
  communicator                  = "winrm"
  image_offer                   = "WindowsServer"
  image_publisher               = "MicrosoftWindowsServer"
  image_sku                      = "2016-Datacenter"
  managed_image_name             = "myPackerImage"
  managed_image_resource_group_name = "myPackerGroup"
  os_type                        = "Windows"
  subscription_id                = "yyyyyyy-yyyy-yyyy-yyyy-yyyyyyyyyy"
  tenant_id                      = "zzzzzz-zzzz-zzzz-zzzz-zzzzzzzzzz"
  vm_size                        = "Standard_D2_v2"
  winrm_insecure                 = true
  winrm_timeout                  = "5m"
  winrm_use_ssl                  = true
  winrm_username                 = "packer"
}

build {
  sources = ["source.azure-arm.autogenerated_1"]

  provisioner "powershell" {
    inline = ["Add-WindowsFeature Web-Server", "while ((Get-Service RdAgent).Status -ne 'Running') { Start-Sleep -s 5 }", "while ((Get-Service WindowsAzureGuestAgent).Status -ne 'Running') { Start-Sleep -s 5 }", "& $env:SystemRoot\\System32\\Sysprep\\Sysprep.exe /oobe /generalize /quiet /quit", "while($true) { $imageState = Get-ItemProperty HKLM:\\SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Setup\\State | Select ImageState; if($imageState.ImageState -ne 'IMAGE_STATE_GENERALIZE_RESEAL_TO_OOBE') { Write-Output $imageState.ImageState; Start-Sleep -s 10 } else { break } }"]
  }
}

```

This template builds a Windows Server 2016 VM, installs IIS, then generalizes the VM with Sysprep. The IIS install shows how you can use the PowerShell provisioner to run additional commands. The final Packer image then includes the required software install and configuration.

The Windows Guest Agent participates in the Sysprep process. The agent must be fully installed before the VM can be sysprep'ed. To ensure that this is true, all agent services must be running before you execute sysprep.exe. The preceding JSON snippet shows one way to do this in the PowerShell provisioner. This snippet is required only if the VM is configured to install the agent, which is the default.

## Build Packer image

If you don't already have Packer installed on your local machine, [follow the Packer installation instructions](#).

Build the image by opening a cmd prompt and specifying your Packer template file as follows:

```
./packer build windows.json
```

You can also build the image by specifying the *windows.pkr.hcl* file as follows:

```
packer build windows.pkr.hcl
```

An example of the output from the preceding commands is as follows:

```

azure-arm output will be in this color.

==> azure-arm: Running builder ...
    azure-arm: Creating Azure Resource Manager (ARM) client ...
==> azure-arm: Creating resource group ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> Location       : 'East US'
==> azure-arm: -> Tags         :
==> azure-arm: ->> task : Image deployment
==> azure-arm: ->> dept : Engineering
==> azure-arm: Validating deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> DeploymentName   : 'pkrdppq0mthbt'
==> azure-arm: Deploying deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> DeploymentName   : 'pkrdppq0mthbt'
==> azure-arm: Getting the certificate's URL ...
==> azure-arm: -> Key Vault Name      : 'pkrvpq0mthbt'
==> azure-arm: -> Key Vault Secret Name : 'packerKeyVaultSecret'
==> azure-arm: -> Certificate URL     :
'https://pkrvpq0mthbt.vault.azure.net/secrets/packerKeyVaultSecret/8c7bd823e4fa44e1abb747636128adbb'
==> azure-arm: Setting the certificate's URL ...
==> azure-arm: Validating deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> DeploymentName   : 'pkrdppq0mthbt'
==> azure-arm: Deploying deployment template ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> DeploymentName   : 'pkrdppq0mthbt'
==> azure-arm: Getting the VM's IP address ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> PublicIPAddressName : 'packerPublicIP'
==> azure-arm: -> NicName        : 'packerNic'
==> azure-arm: -> Network Connection : 'PublicEndpoint'
==> azure-arm: -> IP Address      : '40.76.55.35'
==> azure-arm: Waiting for WinRM to become available...
==> azure-arm: Connected to WinRM!
==> azure-arm: Provisioning with Powershell...
==> azure-arm: Provisioning with shell script: /var/folders/h1/ymh5bxd15wgdn5hvgj1wc0zh0000gn/T/packer-
powershell-provisioner902510110
    azure-arm: #< CLIXML
    azure-arm:
    azure-arm: Success Restart Needed Exit Code      Feature Result
    azure-arm: ----- -----
    azure-arm: True    No           Success      {Common HTTP Features, Default Document, D...
    azure-arm: <Objs Version="1.1.0.1" xmlns="http://schemas.microsoft.com/powershell/2004/04"><Obj
S="progress" RefId="0"><TN RefId="0"><T>System.Management.Automation.PSCustomObject</T><T>System.Object</T>
</TN><MS><I64 N="SourceId">1</I64><PR N="Record"><AV>Preparing modules for first use.</AV><AI>0</AI><Nil />
<PI>-1</PI><PC>-1</PC><T>Completed</T><SR>-1</SR><SD> </SD></PR></MS></Obj></Objs>
==> azure-arm: Querying the machine's properties ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> ComputeName       : 'pkrvmpq0mthbt'
==> azure-arm: -> Managed OS Disk  : '/subscriptions/guid/resourceGroups/packer-Resource-Group-
pq0mthbt/providers/Microsoft.Compute/disks/osdisk'
==> azure-arm: Powering off machine ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> ComputeName       : 'pkrvmpq0mthbt'
==> azure-arm: Capturing image ...
==> azure-arm: -> Compute ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: -> Compute Name        : 'pkrvmpq0mthbt'
==> azure-arm: -> Compute Location     : 'East US'
==> azure-arm: -> Image ResourceGroupName : 'myResourceGroup'
==> azure-arm: -> Image Name         : 'myPackerImage'
==> azure-arm: -> Image Location      : 'eastus'
==> azure-arm: Deleting resource group ...
==> azure-arm: -> ResourceGroupName : 'packer-Resource-Group-pq0mthbt'
==> azure-arm: Deleting the temporary OS disk ...
==> azure-arm: -> OS Disk : skipping, managed disk was used...
Build 'azure-arm' finished.

```

```
==> Builds finished. The artifacts of successful builds are:  
--> azure-arm: Azure.ResourceManagement.VMImage:  
  
ManagedImageResourceGroupName: myResourceGroup  
ManagedImageName: myPackerImage  
ManagedImageLocation: eastus
```

It takes a few minutes for Packer to build the VM, run the provisioners, and clean up the deployment.

## Create a VM from the Packer image

You can now create a VM from your Image with [New-AzVM](#). The supporting network resources are created if they do not already exist. When prompted, enter an administrative username and password to be created on the VM. The following example creates a VM named *myVM* from *myPackerImage*.

```
New-AzVm `  
-ResourceGroupName $rgName `  
-Name "myVM" `  
-Location $location `  
-VirtualNetworkName "myVnet" `  
-SubnetName "mySubnet" `  
-SecurityGroupName "myNetworkSecurityGroup" `  
-PublicIpAddressName "myPublicIpAddress" `  
-OpenPorts 80 `  
-Image "myPackerImage"
```

If you wish to create VMs in a different resource group or region than your Packer image, specify the image ID rather than image name. You can obtain the image ID with [Get-AzImage](#).

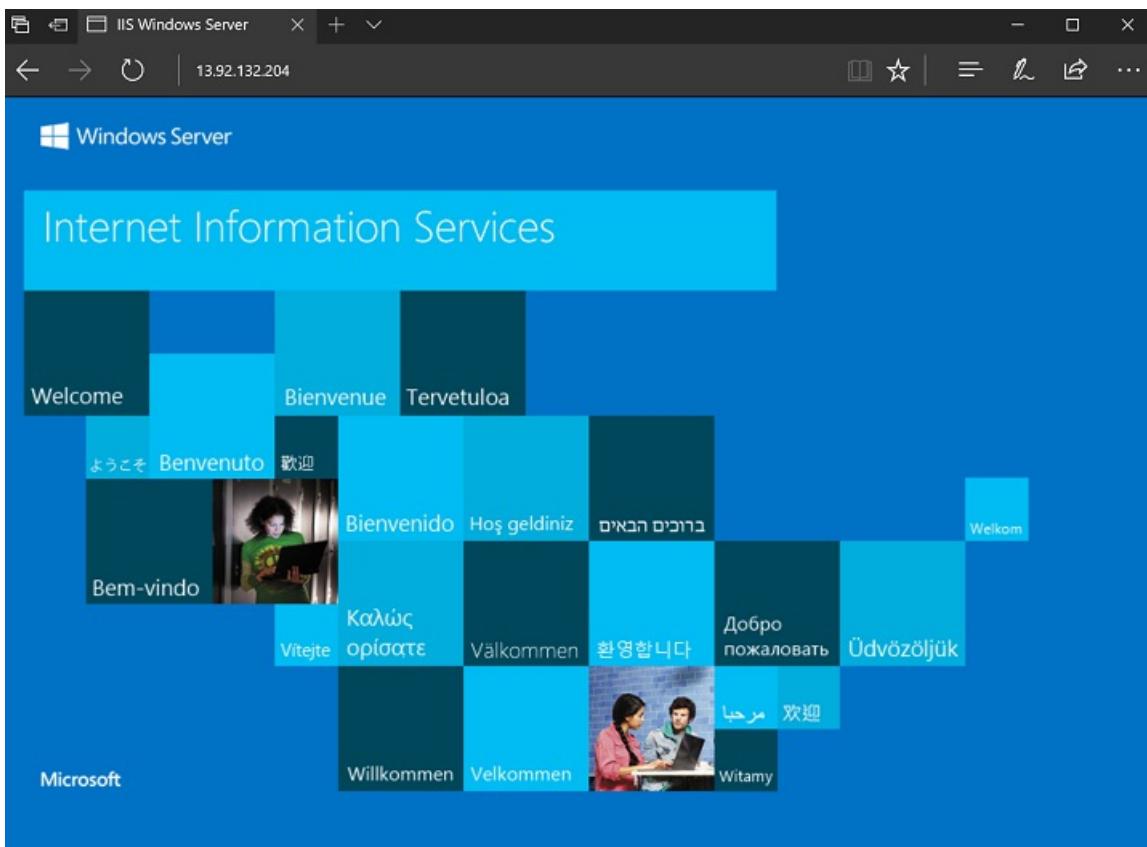
It takes a few minutes to create the VM from your Packer image.

## Test VM and webserver

Obtain the public IP address of your VM with [Get-AzPublicIPAddress](#). The following example obtains the IP address for *myPublicIP* created earlier:

```
Get-AzPublicIPAddress `  
-ResourceGroupName $rgName `  
-Name "myPublicIPAddress" | select "IpAddress"
```

To see your VM, that includes the IIS install from the Packer provisioner, in action, enter the public IP address in to a web browser.



## Next steps

You can also use existing Packer provisioner scripts with [Azure Image Builder](#).

# Azure Dedicated Hosts

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale sets

Azure Dedicated Host is a service that provides physical servers - able to host one or more virtual machines - dedicated to one Azure subscription. Dedicated hosts are the same physical servers used in our data centers, provided as a resource. You can provision dedicated hosts within a region, availability zone, and fault domain. Then, you can place VMs directly into your provisioned hosts, in whatever configuration best meets your needs.

## Benefits

Reserving the entire host provides the following benefits:

- Hardware isolation at the physical server level. No other VMs will be placed on your hosts. Dedicated hosts are deployed in the same data centers and share the same network and underlying storage infrastructure as other, non-isolated hosts.
- Control over maintenance events initiated by the Azure platform. While the majority of maintenance events have little to no impact on your virtual machines, there are some sensitive workloads where each second of pause can have an impact. With dedicated hosts, you can opt in to a maintenance window to reduce the impact to your service.
- With the Azure hybrid benefit, you can bring your own licenses for Windows and SQL to Azure. Using the hybrid benefits provides you with additional benefits. For more information, see [Azure Hybrid Benefit](#).

## Groups, hosts, and VMs



A **host group** is a resource that represents a collection of dedicated hosts. You create a host group in a region and an availability zone, and add hosts to it.

A **host** is a resource, mapped to a physical server in an Azure data center. The physical server is allocated when the host is created. A host is created within a host group. A host has a SKU describing which VM sizes can be created. Each host can host multiple VMs, of different sizes, as long as they are from the same size series.

## High Availability considerations

For high availability, you should deploy multiple VMs, spread across multiple hosts (minimum of 2). With Azure Dedicated Hosts, you have several options to provision your infrastructure to shape your fault isolation boundaries.

### Use Availability Zones for fault isolation

Availability zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. A host group is created in a single availability zone. Once created, all hosts will be placed within that zone. To achieve high availability across zones,

you need to create multiple host groups (one per zone) and spread your hosts accordingly.

If you assign a host group to an availability zone, all VMs created on that host must be created in the same zone.

### Use Fault Domains for fault isolation

A host can be created in a specific fault domain. Just like VM in a scale set or availability set, hosts in different fault domains will be placed on different physical racks in the data center. When you create a host group, you are required to specify the fault domain count. When creating hosts within the host group, you assign fault domain for each host. The VMs do not require any fault domain assignment.

Fault domains are not the same as colocation. Having the same fault domain for two hosts does not mean they are in proximity with each other.

Fault domains are scoped to the host group. You should not make any assumption on anti-affinity between two host groups (unless they are in different availability zones).

VMs deployed to hosts with different fault domains, will have their underlying managed disks services on multiple storage stamps, to increase the fault isolation protection.

### Using Availability Zones and Fault Domains

You can use both capabilities together to achieve even more fault isolation. In this case, you will specify the availability zone and fault domain count in for each host group, assign a fault domain to each of your hosts in the group, and assign an availability zone to each of your VMs

The [Resource Manager sample template](#) uses zones and fault domains to spread hosts for maximum resiliency in a region.

## Manual vs. automatic placement

When creating a VM in Azure, you can select which dedicated host to use. You can also use the option to automatically place your VMs on existing hosts, within a host group.

When creating a new host group, make sure the setting for automatic VM placement is selected. When creating your VM, select the host group and let Azure pick the best host for your VM.

Host groups that are enabled for automatic placement do not require all the VMs to be automatically placed. You will still be able to explicitly pick a host, even when automatic placement is selected for the host group.

### Limitations

Known issues and limitations when using automatic VM placement:

- You will not be able to redeploy your VM.
- You will not be able to use DCv2, Lsv2, NVasv4, NVsv3, Msv2, or M-series VMs with dedicated hosts

## Virtual machine scale set support

Virtual machine scale sets let you treat a group of virtual machines as a single resource, and apply availability, management, scaling and orchestration policies as a group. Your existing dedicated hosts can also be used for virtual machine scale sets.

When creating a virtual machine scale set you can specify an existing host group to have all of the VM instances created on dedicated hosts.

The following requirements apply when creating a virtual machine scale set in a dedicated host group:

- Automatic VM placement needs to be enabled.
- The availability setting of your host group should match your scale set.
  - A regional host group (created without specifying an availability zone) should be used for regional

scale sets.

- The host group and the scale set must be using the same availability zone.
- The fault domain count for the host group level should match the fault domain count for your scale set. The Azure portal lets you specify *max spreading* for your scale set, which sets the fault domain count of 1.
- Dedicated hosts should be created first, with sufficient capacity, and the same settings for scale set zones and fault domains.
- The supported VM sizes for your dedicated hosts should match the one used for your scale set.

Not all scale-set orchestration and optimizations settings are supported by dedicated hosts. Apply the following settings to your scale set:

- Overprovisioning is not recommended, and it is disabled by default. You can enable overprovisioning, but the scale set allocation will fail if the host group does not have capacity for all of the VMs, including the overprovisioned instances.
- Use the ScaleSetVM orchestration mode
- Do not use proximity placement groups for co-location

## Maintenance control

The infrastructure supporting your virtual machines may occasionally be updated to improve reliability, performance, security, and to launch new features. The Azure platform tries to minimize the impact of platform maintenance whenever possible, but customers with *maintenance sensitive* workloads can't tolerate even few seconds that the VM needs to be frozen or disconnected for maintenance.

**Maintenance Control** provides customers with an option to skip regular platform updates scheduled on their dedicated hosts, then apply it at the time of their choice within a 35-day rolling window. Within the maintenance window, you can apply maintenance directly at the host level, in any order. Once the maintenance window is over, Microsoft will move forward and apply the pending maintenance to the hosts in an order which may not follow the user defined fault domains.

For more information, see [Managing platform updates with Maintenance Control](#).

## Capacity considerations

Once a dedicated host is provisioned, Azure assigns it to physical server. This guarantees the availability of the capacity when you need to provision your VM. Azure uses the entire capacity in the region (or zone) to pick a physical server for your host. It also means that customers can expect to be able to grow their dedicated host footprint without the concern of running out of space in the cluster.

## Quotas

There are two types of quota that are consumed when you deploy a dedicated host.

1. Dedicated host vCPU quota. The default quota limit is 3000 vCPUs, per region.
2. VM size family quota. For example, a **Pay-as-you-go** subscription may only have a quota of 10 vCPUs available for the Dsv3 size series, in the East US region. To deploy a Dsv3 dedicated host, you would need to request a quota increase to at least 64 vCPUs before you can deploy the dedicated host.

To request a quota increase, create a support request in the [Azure portal](#).

Provisioning a dedicated host will consume both dedicated host vCPU and the VM family vCPU quota, but it will not consume the regional vCPU. VMs placed on a dedicated host will not count against VM family vCPU quota. Should a VM be moved off a dedicated host into a multi-tenant environment, the VM will consume VM family vCPU quota.

The screenshot shows the Azure portal's 'Pay-As-You-Go - Usage + quotas' section. On the left, there's a sidebar with options like 'Billing', 'Invoices', 'External services', 'Payment methods', 'Partner information', 'Programmatic deployment', 'Resource groups', 'Resources', and 'Usage + quotas'. The 'Usage + quotas' option is highlighted. The main area has a search bar and a 'Refresh' button. It displays a message about quotas and usage per subscription. Below that, there are dropdown menus for 'All service quotas', 'Microsoft.Compute', and 'East US', and a 'Show all' button. A search bar contains 'dsv3'. A table header includes 'QUOTA', 'PROVIDER', 'LOCATION', and 'USAGE'. A single row is shown: 'Standard Dsv3 Family vCPUs' under 'Microsoft.Compute' in 'East US', with a usage bar at 0% and '0 of 10'.

For more information, see [Virtual machine vCPU quotas](#).

Free trial and MSDN subscriptions do not have quota for Azure Dedicated Hosts.

## Pricing

Users are charged per dedicated host, regardless how many VMs are deployed. In your monthly statement you will see a new billable resource type of hosts. The VMs on a dedicated host will still be shown in your statement, but will carry a price of 0.

The host price is set based on VM family, type (hardware size), and region. A host price is relative to the largest VM size supported on the host.

Software licensing, storage and network usage are billed separately from the host and VMs. There is no change to those billable items.

For more information, see [Azure Dedicated Host pricing](#).

You can also save on costs with a [Reserved Instance of Azure Dedicated Hosts](#).

## Sizes and hardware generations

A SKU is defined for a host and it represents the VM size series and type. You can mix multiple VMs of different sizes within a single host as long as they are of the same size series.

The *type* is the hardware generation. Different hardware types for the same VM series will be from different CPU vendors and have different CPU generations and number of cores.

The sizes and hardware types vary by region. Refer to the host [pricing page](#) to learn more.

### NOTE

Once a Dedicated host is provisioned, you can't change the size or type. If you need a different size or type, you will need to create a new host.

## Host life cycle

Azure monitors and manages the health status of your hosts. The following states will be returned when you query your host:

HEALTH STATE	DESCRIPTION
Host Available	There are no known issues with your host.

Health State	Description
Host Under Investigation	We're having some issues with the host which we're looking into. This is a transitional state required for Azure to try and identify the scope and root cause for the issue identified. Virtual machines running on the host may be impacted.
Host Pending Deallocate	Azure can't restore the host back to a healthy state and ask you to redeploy your virtual machines out of this host. If <code>autoReplaceOnFailure</code> is enabled, your virtual machines are <i>service healed</i> to healthy hardware. Otherwise, your virtual machine may be running on a host that is about to fail.
Host deallocated	All virtual machines have been removed from the host. You are no longer being charged for this host since the hardware was taken out of rotation.

## Next steps

- To deploy a dedicated host, see [Deploy VMs and scale sets to dedicated hosts](#).
- There is a [sample template](#) that uses both zones and fault domains for maximum resiliency in a region.
- You can also save on costs with a [Reserved Instance of Azure Dedicated Hosts](#).

# Deploy VMs and scale sets to dedicated hosts

9/21/2022 • 16 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale sets

This article guides you through how to create an Azure [dedicated host](#) to host your virtual machines (VMs) and scale set instances.

## Limitations

- The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.
- Not all Azure VM SKUs, regions and availability zones support ultra disks, for more information about this topic, see [Azure ultra disks](#).
- The fault domain count of the virtual machine scale set can't exceed the fault domain count of the host group.

## Create a host group

A **host group** is a resource that represents a collection of dedicated hosts. You create a host group in a region and an availability zone, and add hosts to it. You can use one or both of the following options with your dedicated hosts to ensure high availability:

- Span across multiple availability zones. In this case, you're required to have a host group in each of the zones you wish to use.
- Span across multiple fault domains, which are mapped to physical racks.

In either case, you need to provide the fault domain count for your host group. If you don't want to span fault domains in your group, use a fault domain count of 1.

You can also decide to use both availability zones and fault domains.

Enabling ultra disks is a host group level setting and can't be changed after a host group is created.

If you intend to use LSv2 or M series VMs, with ultra disks on dedicated hosts, set host group's **Fault domain count** to 1.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

In this example, we'll create a host group using one availability zone and two fault domains.

1. Open the Azure [portal](#).
2. Select **Create a resource** in the upper left corner.
3. Search for **Host group** and then select **Host Groups** from the results.
4. In the **Host Groups** page, select **Create**.
5. Select the subscription you would like to use, and then select **Create new** to create a new resource group.
6. Type *myDedicatedHostsRG* as the **Name** and then select **OK**.
7. For **Host group name**, type *myHostGroup*.

8. For **Location**, select **East US**.
9. For **Availability Zone**, select **1**.
10. Select **Enable Ultra SSD** to use ultra disks with supported Virtual Machines.
11. For **Fault domain count**, select **2**.
12. Select **Automatic placement** to automatically assign VMs and scale set instances to an available host in this group.
13. Select **Review + create** and then wait for validation.
14. Once you see the **Validation passed** message, select **Create** to create the host group.

It should only take a few moments to create the host group.

## Create a dedicated host

Now create a dedicated host in the host group. In addition to a name for the host, you're required to provide the SKU for the host. Host SKU captures the supported VM series and the hardware generation for your dedicated host.

For more information about the host SKUs and pricing, see [Azure Dedicated Host pricing](#).

If you set a fault domain count for your host group, you'll need to specify the fault domain for your host.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Select **Create a resource** in the upper left corner.
2. Search for **Dedicated host** and then select **Dedicated hosts** from the results.
3. In the **Dedicated Hosts** page, select **Create**.
4. Select the subscription you would like to use.
5. Select *myDedicatedHostsRG* as the **Resource group**.
6. In **Instance details**, type *myHost* for the **Name** and select *East US* for the location.
7. In **Hardware profile**, select *Standard Es3 family - Type 1* for the **Size family**, select *myHostGroup* for the **Host group** and then select *1* for the **Fault domain**. Leave the defaults for the rest of the fields.
8. When you're done, select **Review + create** and wait for validation.
9. Once you see the **Validation passed** message, select **Create** to create the host.

## Create a VM

Now create a VM on the host.

If you would like to create a VM with ultra disks support, make sure the host group in which the VM will be placed is ultra SSD enabled. Once you've confirmed, create the VM in the same host group. See [Deploy an ultra disk](#) for the steps to attach an ultra disk to a VM.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Choose **Create a resource** in the upper left corner of the Azure portal.
2. In the search box above the list of Azure Marketplace resources, search for and select the image you want to use, then choose **Create**.
3. In the **Basics** tab, under **Project details**, make sure the correct subscription is selected and then select *myDedicatedHostsRG* as the **Resource group**.

4. Under **Instance details**, type *myVM* for the **Virtual machine name** and choose *East US* for your **Location**.
5. In **Availability options** select **Availability zone**, select *1* from the drop-down.
6. For the size, select **Change size**. In the list of available sizes, choose one from the Esv3 series, like **Standard E2s v3**. You may need to clear the filter in order to see all of the available sizes.
7. Complete the rest of the fields on the **Basics** tab as needed.
8. If you want to specify which host to use for your VM, then at the top of the page, select the **Advanced** tab and in the **Host** section, select *myHostGroup* for **Host group** and *myHost* for the **Host**. Otherwise, your VM will automatically be placed on a host with capacity.

**Host**

Optionally placing your virtual machine in a host [Learn more](#)

Host group	myHostGroup   Zone 1   eastus	
Host	myHost	

9. Leave the remaining defaults and then select the **Review + create** button at the bottom of the page.
10. When you see the message that validation has passed, select **Create**.

It will take a few minutes for your VM to be deployed.

## Create a scale set

You can also create a scale set on your host.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

When you deploy a scale set, you specify the host group.

1. Search for **Scale set** and select **Virtual machine scale sets** from the list.
2. Select **Add** to create a new scale set.
3. Complete the fields on the **Basics** tab as you usually would, but make sure you select a VM size that is from the series you chose for your dedicated host, like **Standard E2s v3**.
4. On the **Advanced** tab, for **Spreading algorithm** select **Max spreading**.
5. In **Host group**, select the host group from the drop-down. If you recently created the group, it might take a minute to get added to the list.

## Add an existing VM

You can add an existing VM to a dedicated host, but the VM must first be Stop\Deallocated. Before you move a VM to a dedicated host, make sure that the VM configuration is supported:

- The VM size must be in the same size family as the dedicated host. For example, if your dedicated host is DSv3, then the VM size could be Standard\_D4s\_v3, but it couldn't be a Standard\_A4\_v2.
- The VM needs to be located in same region as the dedicated host.
- The VM can't be part of a proximity placement group. Remove the VM from the proximity placement group before moving it to a dedicated host. For more information about this topic, see [Move a VM out of a proximity placement group](#)
- The VM can't be in an availability set.
- If the VM is in an availability zone, it must be the same availability zone as the host group. The availability zone settings for the VM and the host group must match.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

Move the VM to a dedicated host using the [portal](#).

1. Open the page for the VM.
2. Select **Stop** to stop\deallocate the VM.
3. Select **Configuration** from the left menu.
4. Select a host group and a host from the drop-down menus.
5. When you're done, select **Save** at the top of the page.
6. After the VM has been added to the host, select **Overview** from the left menu.
7. At the top of the page, select **Start** to restart the VM.

## Check the status of the host

If you need to know how much capacity is still available on a host, you can check the status.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Search for and select the host.
2. In the **Overview** page for the host, scroll down to see the list of sizes still available for the host. It should look similar to:

---

### Available VM capacity

VM Size	Number remaining
Standard_D2ds_v4	32
Standard_D4ds_v4	17
Standard_D8ds_v4	8
Standard_D16ds_v4	4
Standard_D32ds_v4	2
Standard_D48ds_v4	1
Standard_D64ds_v4	1

## Restart a host (Preview)

You can restart the entire host, meaning that the host's not completely powered off. Because the host will be restarted, the underlying VMs will also be restarted. The host will remain on the same underlying physical hardware as it restarts and both the host ID and asset ID will remain the same after the restart. The host SKU will also remain the same after the restart.

Note: Host restart is in preview.

- [Portal](#)

- [CLI](#)
- [PowerShell](#)

1. Search for and select the host.
2. In the top menu bar, select the **Restart** button. Note, this feature is in Preview.
3. In the **Essentials** section of the Host Resource Pane, Host Status will switch to **Host undergoing restart** during the restart.
4. Once the restart has completed, the Host Status will return to **Host available**.

## Deleting a host

You're being charged for your dedicated host even when no virtual machines are deployed on the host. You should delete any hosts you're currently not using to save costs.

You can only delete a host when there are no any longer virtual machines using it.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Search for and select the host.
2. In the left menu, select **Instances**.
3. Select and delete each virtual machine.
4. When all of the VMs have been deleted, go back to the **Overview** page for the host and select **Delete** from the top menu.
5. Once the host has been deleted, open the page for the host group and select **Delete host group**.

## Next steps

- For more information about this topic, see the [Dedicated hosts](#) overview.
- There's sample template, available at [Azure Quickstart Templates](#), which uses both zones and fault domains for maximum resiliency in a region.

# General Purpose Azure Dedicated Host SKUs

9/21/2022 • 10 minutes to read • [Edit Online](#)

Azure Dedicated Host SKUs are the combination of a VM family and a certain hardware specification. You can only deploy VMs of the VM series that the Dedicated Host SKU specifies. For example, on the Dsv3-Type3, you can only provision [Dsv3-series](#) VMs.

This document goes through the hardware specifications and VM packings for all general purpose Dedicated Host SKUs.

## Limitations

The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.

## Dadsv5

### Dadsv5-Type1

The Dadsv5-Type1 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Dadsv5-Type1 runs [Dadsv5-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dadsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	D2ads v5	32
			D4ads v5	27
			D8ads v5	14
			D16ads v5	7
			D32ads v5	3
			D48ads v5	2
			D64ads v5	1
			D96ads v5	1

## Dasv5

### Dasv5-Type1

The Dasv5-Type1 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Dasv5-Type1 runs [Dasv5-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dasv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	D2as v5	32
			D4as v5	28
			D8as v5	14
			D16as v5	7
			D32as v5	3
			D48as v5	2
			D64as v5	1
			D96as v5	1

## Dasv4

### **Dasv4-Type1**

The Dasv4-Type1 is a Dedicated Host SKU utilizing AMD's 2.35 GHz EPYC™ 7452 processor. It offers 64 physical cores, 96 vCPUs, and 672 GiB of RAM. The Dasv4-Type1 runs [Dasv4-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dasv4-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	96	672 GiB	D2as v4	32
			D4as v4	24
			D8as v4	12
			D16as v4	6
			D32as v4	3
			D48as v4	2
			D64as v4	1
			D96as v4	1

You can also mix multiple VM sizes on the Dasv4-Type1. The following are sample combinations of VM packings on the Dasv4-Type1:

- 1 D48asv4 + 3 D16asv4

- 1 D32asv4 + 2 D16asv4 + 8 D4asv4
- 20 D4asv4 + 8 D2asv4

### **Dasv4-Type2**

The Dasv4-Type2 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Dasv4-Type2 runs [Dasv4-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dasv4-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	D2as v4	32
			D4as v4	25
			D8as v4	12
			D16as v4	6
			D32as v4	3
			D48as v4	2
			D64as v4	1
			D96as v4	1

## **DCadsv5**

### **DCadsv5-Type1**

The DCadsv5-Type1 is a Dedicated Host SKU utilizing the AMD 3rd Generation EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The DCadsv5-Type1 runs [DCadsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an DCadsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	DC2ads v5	32
			DC4ads v5	27
			DC8ads v5	14
			DC16ads v5	7
			DC32ads v5	3
			DC48ads v5	2
			DC64ads v5	1

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			DC96ads v5	1

## DCasv5

### DCasv5-Type1

The DCasv5-Type1 is a Dedicated Host SKU utilizing the AMD 3rd Generation EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The DCasv5-Type1 runs [DCasv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an DCasv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	DC2as v5	32
			DC4as v5	28
			DC8as v5	14
			DC16as v5	7
			DC32as v5	3
			DC48as v5	2
			DC64as v5	1
			DC96as v5	1

## DCsv3

### DCsv3-Type1

The DCsv3-Type1 is a Dedicated Host SKU utilizing the 3rd Generation Intel® Xeon Scalable Processor 8370C. It offers 48 physical cores, 48 vCPUs, and 384 GiB of RAM. The DCsv3-Type1 runs [DCsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a DCsv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
48	48	384 GiB	DC1s v3	32
			DC2s v3	24
			DC4s v3	12
			DC8s v3	6
			DC16s v3	3

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			DC24s v3	2

## DCcsv3

### DCcsv3-Type1

The DCcsv3-Type1 is a Dedicated Host SKU utilizing the 3rd Generation Intel® Xeon Scalable Processor 8370C. It offers 48 physical cores, 48 vCPUs, and 384 GiB of RAM. The DCcsv3-Type1 runs [DCcsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a DCcsv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
48	48	384 GiB	DC16ds v3	2
			DC24ds v3	1
			DC32ds v3	1
			DC48ds v3	1

## DCsv2

### DCsv2-Type1

The DCsv2-Type1 is a Dedicated Host SKU utilizing the Intel® Coffee Lake (Xeon® E-2288G with SGX technology) processor. It offers 8 physical cores, 8 vCPUs, and 64 GiB of RAM. The DCsv2-Type1 runs [DCsv2-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a DCsv2-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
8	8	64 GiB	DC8 v2	1

## Ddsv5

### Ddsv5-Type1

The Ddsv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Ddsv5-Type1 runs [Ddsv5-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Ddsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	D2ds v5	32
			D4ds v5	22

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
			D8ds v5	11
			D16ds v5	5
			D32ds v5	2
			D48ds v5	1
			D64ds v5	1
			D96ds v5	1

## Ddsv4

### Ddsv4-Type1

The Ddsv4-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Ddsv4-Type1 runs [Ddsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Ddsv4-Type1 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
52	80	504 GiB	D2ds v4	32
			D4ds v4	17
			D8ds v4	8
			D16ds v4	4
			D32ds v4	2
			D48ds v4	1
			D64ds v4	1

You can also mix multiple VM sizes on the Ddsv4-Type1. The following are sample combinations of VM packings on the Ddsv4-Type1:

- 1 D48dsv4 + 4 D4dsv4 + 2 D2dsv4
- 1 D32dsv4 + 2 D16dsv4 + 1 D4dsv4
- 10 D4dsv4 + 14 D2dsv4

### Ddsv4-Type2

The Ddsv4-Type2 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Ddsv4-Type2 runs [Ddsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Ddsv4-Type2

host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	D2ds v4	32
			D4ds v4	19
			D8ds v4	9
			D16ds v4	4
			D32ds v4	2
			D48ds v4	1
			D64ds v4	1

## Dsv5

### Dsv5-Type1

The Dsv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Dsv5-Type1 runs [Dsv5-series](#) VMs. Refer to the VM size documentation to better understand specific VM performance information.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	D2s v5	32
			D4s v5	25
			D8s v5	12
			D16s v5	6
			D32s v5	3
			D48s v5	2
			D64s v5	1
			D96s v5	1

## Dsv4

### Dsv4-Type1

The Dsv4-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Dsv4-Type1 runs [Dsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv4-Type1

host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	D2s v4	32
			D4s v4	20
			D8s v4	10
			D16s v4	5
			D32s v4	2
			D48s v4	1
			D64s v4	1

You can also mix multiple VM sizes on the Dsv4-Type1. The following are sample combinations of VM packings on the Dsv4-Type1:

- 1 D64sv4 + 1 D16sv4
- 1 D32sv4 + 2 D16sv4 + 2 D8sv4
- 10 D4sv4 + 20 D2sv4

## Dsv4-Type2

The Dsv4-Type2 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Dsv4-Type2 runs [Dsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv4-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	D2s v4	32
			D4s v4	25
			D8s v4	12
			D16s v4	6
			D32s v4	3
			D48s v4	2
			D64s v4	1

## Dsv3

### Dsv3-Type1

**NOTE**

The Dsv3-Type1 will be retired on March 31, 2023. Refer to the [dedicated host retirement guide](#) to learn more.

The Dsv3-Type1 is a Dedicated Host SKU utilizing the Intel® Broadwell (2.3 GHz Xeon® E5-2673 v4) processor. It offers 40 physical cores, 64 vCPUs, and 256 GiB of RAM. The Dsv3-Type1 runs [Dsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
40	64	256 GiB	D2s v3	32
			D4s v3	17
			D8s v3	8
			D16s v3	4
			D32s v3	2
			D48s v3	1
			D64s v3	1

You can also mix multiple VM sizes on the Dsv3-Type1. The following are sample combinations of VM packings on the Dsv3-Type1:

- 1 D32sv3 + 1 D16sv3 + 1 D8sv3
- 1 D48sv3 + 3 D4sv3 + 2 D2sv3
- 10 D4sv3 + 12 D2sv3

## Dsv3-Type2

**NOTE**

The Dsv3-Type2 will be retired on March 31, 2023. Refer to the [dedicated host retirement guide](#) to learn more.

The Dsv3-Type2 is a Dedicated Host SKU utilizing the Intel® Skylake (2.1 GHz Xeon® Platinum 8171M) processor. It offers 48 physical cores, 76 vCPUs, and 504 GiB of RAM. The Dsv3-Type2 runs [Dsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv3-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
48	76	504 GiB	D2s v3	32
			D4s v3	20
			D8s v3	10

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			D16s v3	5
			D32s v3	2
			D48s v3	1
			D64s v3	1

You can also mix multiple VM sizes on the Dsv3-Type2. The following are sample combinations of VM packings on the Dsv3-Type2:

- 1 D64sv3 + 2 D4sv3 + 2 D2sv3
- 1 D48sv3 + 4 D4sv3 + 6 D2sv3
- 12 D4sv3 + 14 D2sv3

### Dsv3-Type3

The Dsv3-Type3 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Dsv3-Type3 runs [Dsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv3-Type3 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	D2s v3	32
			D4s v3	21
			D8s v3	10
			D16s v3	5
			D32s v3	2
			D48s v3	1
			D64s v3	1

You can also mix multiple VM sizes on the Dsv3-Type3. The following are sample combinations of VM packings on the Dsv3-Type3:

- 1 D64sv3 + 1 D8sv3 + 2 D4sv3
- 1 D48sv3 + 1 D16sv3 + 4 D4sv3
- 15 D4sv3 + 10 D2sv3

### Dsv3-Type4

The Dsv3-Type4 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Dsv3-Type4 runs [Dsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Dsv3-Type4 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
64	119	768 GiB	D2s v3	32
			D4s v3	24
			D8s v3	12
			D16s v3	6
			D32s v3	3
			D48s v3	2
			D64s v3	1

## Next steps

- For more information, see the [Dedicated hosts](#) overview.
- There is sample template, available at [Azure Quickstart Templates](#) that uses both zones and fault domains for maximum resiliency in a region.

# Compute Optimized Azure Dedicated Host SKUs

9/21/2022 • 2 minutes to read • [Edit Online](#)

Azure Dedicated Host SKUs are the combination of a VM family and a certain hardware specification. You can only deploy VMs of the VM series that the Dedicated Host SKU specifies. For example, on the Dsv3-Type3, you can only provision [Dsv3-series](#) VMs.

This document goes through the hardware specifications and VM packings for all compute optimized Dedicated Host SKUs.

## Limitations

The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.

## Fsv2

### Fsv2-Type2

The Fsv2-Type2 is a Dedicated Host SKU utilizing the Intel® Skylake (Xeon® Platinum 8168) processor. It offers 48 physical cores, 72 vCPUs, and 144 GiB of RAM. The Fsv2-Type2 runs [Fsv2-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Fsv2-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMs
48	72	144 GiB	F2s v2	32
			F4s v2	18
			F8s v2	9
			F16s v2	4
			F32s v2	2
			F48s v2	1
			F64s v2	1
			F72s v2	1

### Fsv2-Type3

The Fsv2-Type3 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Fsv2-Type3 runs [Fsv2-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Fsv2-Type3 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	F2s v2	32
			F4s v2	21
			F8s v2	10
			F16s v2	5
			F32s v2	2
			F48s v2	1
			F64s v2	1
			F72s v2	1

### Fsv2-Type4

The Fsv2-Type4 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Fsv2-Type4 runs [Fsv2-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Fsv2-Type4 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	F2s v2	32
			F4s v2	24
			F8s v2	12
			F16s v2	6
			F32s v2	3
			F48s v2	2
			F64s v2	1
			F72s v2	1

## FXmds

### FXmds-Type1

The FXmds-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Gold 6246R) processor. It offers 32 physical cores, 48 vCPUs, and 1,152 GiB of RAM. The FXmds-Type1 runs [FX-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a FXmds-Type1 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
32	48	1,152 GiB	FX4mds	12
			FX12mds	4
			FX24mds	2
			FX36mds	1
			FX48mds	1

## Next steps

- For more information, see the [Dedicated hosts](#) overview.
- There is sample template, available at [Azure quickstart templates](#), that uses both zones and fault domains for maximum resiliency in a region.

# Memory Optimized Azure Dedicated Host SKUs

9/21/2022 • 12 minutes to read • [Edit Online](#)

Azure Dedicated Host SKUs are the combination of a VM family and a certain hardware specification. You can only deploy VMs of the VM series that the Dedicated Host SKU specifies. For example, on the Dsv3-Type3, you can only provision [Dsv3-series](#) VMs.

This document goes through the hardware specifications and VM packings for all memory optimized Dedicated Host SKUs.

## Limitations

The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.

## Eadsv5

### **Eadsv5-Type1**

The Eadsv5-Type1 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Eadsv5-Type1 runs [Eadsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Eadsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMs
64	112	768 GiB	E2ads v5	32
			E4ads v5	21
			E8ads v5	10
			E16ads v5	5
			E20ads v5	4
			E32ads v5	2
			E48ads v5	1
			E64ads v5	1
			E96ads v5	1

## Easv5

### **Easv5-Type1**

The Easv5-Type1 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Easv5-Type1 runs [Easv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Easv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	E2as v5	32
			E4as v5	21
			E8as v5	10
			E16as v5	5
			E20as v5	4
			E32as v5	2
			E48as v5	1
			E64as v5	1
			E96as v5	1

## Easv4

### Easv4-Type1

The Easv4-Type1 is a Dedicated Host SKU utilizing AMD's 2.35 GHz EPYC™ 7452 processor. It offers 64 physical cores, 96 vCPUs, and 672 GiB of RAM. The Easv4-Type1 runs [Easv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Easv4-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	96	672 GiB	E2as v4	32
			E4as v4	21
			E8as v4	10
			E16as v4	5
			E20as v4	4
			E32as v4	2
			E48as v4	1
			E64as v4	1
			E96as v4	1

### Easv4-Type2

The Easv4-Type2 is a Dedicated Host SKU utilizing AMD's EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The Easv4-Type2 runs [Easv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Easv4-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	E2as v4	32
			E4as v4	21
			E8as v4	10
			E16as v4	5
			E20as v4	4
			E32as v4	2
			E48as v4	1
			E64as v4	1
			E96as v4	1

## Ebdsv5

### Ebdsv5-Type1

The Ebdsv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Ebdsv5-Type1 runs [Ebdsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Ebdsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2bds v5	8
			E4bds v5	8
			E8bds v5	6
			E16bds v5	3
			E32bds v5	1
			E48bds v5	1
			E64bds v5	1

## Ebsv5

## Ebsv5-Type1

The Ebsv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Ebsv5-Type1 runs [Ebsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Ebsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2bs v5	8
			E4bs v5	8
			E8bs v5	6
			E16bs v5	3
			E32bs v5	1
			E48bs v5	1
			E64bs v5	1

## ECadsv5

### ECadsv5-Type1

The ECadsv5-Type1 is a Dedicated Host SKU utilizing the AMD 3rd Generation EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The ECadsv5-Type1 runs [ECadsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an ECadsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	EC2ads v5	32
			EC4ads v5	21
			EC8ads v5	10
			EC16ads v5	5
			EC20ads v5	4
			EC32ads v5	3
			EC48ads v5	1
			EC64ads v5	1
			EC96ads v5	1

# ECasv5

## ECasv5-Type1

The ECasv5-Type1 is a Dedicated Host SKU utilizing the AMD 3rd Generation EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 768 GiB of RAM. The ECasv5-Type1 runs [ECasv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an ECasv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	112	768 GiB	EC2as v5	32
			EC4as v5	21
			EC8as v5	10
			EC16as v5	5
			EC20as v5	4
			EC32as v5	3
			EC48as v5	1
			EC64as v5	1
			EC96as v5	1

# Edsv5

## Edsv5-Type1

The Edsv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Edsv5-Type1 runs [Edsv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Edsv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2ds v5	32
			E4ds v5	21
			E8ds v5	10
			E16ds v5	5
			E20ds v5	4
			E32ds v5	2
			E48ds v5	1

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			E64ds v5	1

## Edsv4

### Edsv4-Type1

The Edsv4-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Edsv4-Type1 runs [Edsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Edsv4-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	E2ds v4	31
			E4ds v4	15
			E8ds v4	7
			E16ds v4	3
			E20ds v4	3
			E32ds v4	1
			E48ds v4	1
			E64ds v4	1

### Edsv4-Type2

The Edsv4-Type2 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Edsv4-Type2 runs [Edsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Edsv4-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2ds v4	32
			E4ds v4	19
			E8ds v4	9
			E16ds v4	4
			E20ds v4	3

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			E32ds v4	2
			E48ds v4	1
			E64ds v4	1

## Esv5

### Esv5-Type1

The Esv5-Type1 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Esv5-Type1 runs [Esv5-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv5-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2s v5	32
			E4s v5	21
			E8s v5	10
			E16s v5	5
			E20s v5	4
			E32s v5	2
			E48s v5	1
			E64s v5	1

## Esv4

### Esv4-Type1

The Esv4-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Esv4-Type1 runs [Esv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv4-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	E2s v4	31
			E4s v4	15
			E8s v4	7

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			E16s v4	3
			E20s v4	3
			E32s v4	1
			E48s v4	1
			E64s v4	1

### Esv4-Type2

The Esv4-Type2 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Esv4-Type2 runs [Esv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv4-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2s v4	32
			E4s v4	21
			E8s v4	10
			E16s v4	5
			E20s v4	4
			E32s v4	2
			E48s v4	1
			E64s v4	1

## Esv3

### Esv3-Type1

#### NOTE

The Esv3-Type1 will be retired on March 31, 2023. Refer to the [dedicated host retirement guide](#) to learn more.

The Esv3-Type1 is a Dedicated Host SKU utilizing the Intel® Broadwell (2.3 GHz Xeon® E5-2673 v4) processor. It offers 40 physical cores, 64 vCPUs, and 448 GiB of RAM. The Esv3-Type1 runs [Esv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMs
40	64	448 GiB	E2s v3	28
			E4s v3	14
			E8s v3	7
			E16s v3	3
			E20s v3	2
			E32s v3	1
			E48s v3	1
			E64s v3	1

## Esv3-Type2

### NOTE

The Esv3-Type2 will be retired on March 31, 2023. Refer to the [dedicated host retirement guide](#) to learn more.

The Esv3-Type2 is a Dedicated Host SKU utilizing the Intel® Skylake (Xeon® 8171M) processor. It offers 48 physical cores, 76 vCPUs, and 504 GiB of RAM. The Esv3-Type2 runs [Esv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv3-Type2 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMs
48	76	504 GiB	E2s v3	31
			E4s v3	15
			E8s v3	7
			E16s v3	3
			E20s v3	3
			E32s v3	1
			E48s v3	1
			E64s v3	1

## Esv3-Type3

The Esv3-Type3 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8272CL) processor. It offers 52 physical cores, 80 vCPUs, and 504 GiB of RAM. The Esv3-Type3 runs [Esv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv3-Type3

host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
52	80	504 GiB	E2s v3	31
			E4s v3	15
			E8s v3	7
			E16s v3	3
			E20s v3	3
			E32s v3	1
			E48s v3	1
			E64s v3	1

#### Esv3-Type4

The Esv3-Type4 is a Dedicated Host SKU utilizing the Intel® Ice Lake (Xeon® Platinum 8370C) processor. It offers 64 physical cores, 119 vCPUs, and 768 GiB of RAM. The Esv3-Type4 runs [Esv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto an Esv3-Type4 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	768 GiB	E2s v3	32
			E4s v3	21
			E8s v3	10
			E16s v3	5
			E20s v3	4
			E32s v3	2
			E48s v3	1
			E64s v3	1

## M

#### Ms-Type1

The Ms-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 128 vCPUs, and 2,048 GiB of RAM. The Ms-Type1 runs [M-series](#) VMs, including M, MIs, Ms, and Mts VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Ms-Type1

host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	128	2,048 GiB	M32ls	4
			M32ts	4
			M64	2
			M64s	2
			M64ls	2
			M128	1
			M128s	1

### Msm-Type1

The Msm-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 128 vCPUs, and 3,892 GiB of RAM. The Msm-Type1 runs [M-series](#) VMs, including Ms, Mms, Mts, and Mls VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Msm-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	128	3,892 GiB	M8ms	16
			M8-2ms	16
			M8-4ms	16
			M16ms	8
			M16-4ms	8
			M16-8ms	8
			M32ts	4
			M32ls	4
			M32ms	4
			M32-8ms	4
			M32-16ms	4
			M64ms	2
			M64	2

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			M64s	2
			M64m	2
			M64ls	2
			M64-16ms	2
			M64-32ms	2
			M128ms	1
			M128s	1
			M128m	1
			M128	1
			M128-32ms	1
			M128-64ms	1

## Mdsv2

### Mdmsv2MedMem-Type1

The Mdmsv2MedMem-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 192 vCPUs, and 4,096 GiB of RAM. The Mdmsv2MedMem-Type1 runs [Msv2-series](#) VMs, including Mdsv2 and Mdmsv2 VMs.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	192	4,096 GiB	M32dms v2	4
			M64ds v2	2
			M64dms v2	2
			M128ds v2	1
			M128dms v2	1

### Mdsv2MedMem-Type1

The Mdsv2MedMem-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 192 vCPUs, and 2,048 GiB of RAM. The Mdsv2MedMem-Type1 runs [Msv2-series](#) VMs, including Mdsv2 and Mdmsv2 VMs.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	192	2,048 GiB	M32dms v2	2

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
			M64ds v2	2
			M64dms v2	1
			M128ds v2	1

## Msv2

### Mmsv2MedMem-Type1

The Mmsv2MedMem-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 192 vCPUs, and 4,096 GiB of RAM. The Mmsv2MedMem-Type1 runs [Msv2-series](#) VMs, including Msv2 and Mmsv2 VMs.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	192	4,096 GiB	M32ms v2	4
			M64s v2	3
			M64ms v2	2
			M128ms v2	1
			M128s v2	1

### Msv2MedMem-Type1

The Msv2MedMem-Type1 is a Dedicated Host SKU utilizing the Intel® Cascade Lake (Xeon® Platinum 8280) processor. It offers 112 physical cores, 192 vCPUs, and 2,048 GiB of RAM. The Msv2MedMem-Type1 runs [Msv2-series](#) VMs, including Msv2 and Mmsv2 VMs.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
112	192	2,048 GiB	M32ms v2	2
			M64s v2	2
			M64ms v2	1
			M128s v2	1

## Mv2

### Msmv2-Type1

The Msm-Type1 is a Dedicated Host SKU utilizing the Intel® Skylake (Xeon® Platinum 8180M) processor. It offers 224 physical cores, 416 vCPUs, and 11,400 GiB of RAM. The Msmv2-Type1 runs [Mv2-series](#) VMs, including Msv2 and Mmsv2 VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Msm-Type1 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
224	416	11,400 GiB	M208ms v2	2
			M208s v2	2
			M416-208ms v2	1
			M416-208s v2	1
			M416ms v2	1
			M416s v2	1

### Msv2-Type1

The Msv2-Type1 is a Dedicated Host SKU utilizing the Intel® Skylake (Xeon® Platinum 8180M) processor. It offers 224 physical cores, 416 vCPUs, and 5,700 GiB of RAM. The Msv2-Type1 runs [Mv2-series](#) VMs, including Msv2 and Mmsv2 VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Msv2-Type1 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
224	416	5,700 GiB	M208ms v2	2
			M208s v2	1
			M416-208s v2	1
			M416s v2	1

## Next steps

- For more information, see the [Dedicated hosts](#) overview.
- There's sample template, available at [Azure Quickstart Templates](#), which uses both zones and fault domains for maximum resiliency in a region.

# Storage Optimized Azure Dedicated Host SKUs

9/21/2022 • 2 minutes to read • [Edit Online](#)

Azure Dedicated Host SKUs are the combination of a VM family and a certain hardware specification. You can only deploy VMs of the VM series that the Dedicated Host SKU specifies. For example, on the Dsv3-Type3, you can only provision [Dsv3-series](#) VMs.

This document goes through the hardware specifications and VM packings for all storage optimized Dedicated Host SKUs.

## Limitations

The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.

## Lasv3

### Lasv3-Type1

The Lasv3-Type1 is a Dedicated Host SKU utilizing the AMD 3rd Generation EPYC™ 7763v processor. It offers 64 physical cores, 112 vCPUs, and 1024 GiB of RAM. The Lasv3-Type1 runs [Lasv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Lasv3-Type1 host.

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
64	112	1024 GiB	L8as v3	10
			L16as v3	5
			L32as v3	2
			L48as v3	1
			L64as v3	1
			L80as v3	1

## Lsv3

### Lsv3-Type1

The Lsv3-Type1 is a Dedicated Host SKU utilizing the Intel® 3rd Generation Xeon® Platinum 8370C (Ice Lake) processor. It offers 64 physical cores, 119 vCPUs, and 1024 GiB of RAM. The Lsv3-Type1 runs [Lsv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a Lsv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	119	1024 GiB	L8s v3	10
			L16s v3	5
			L32s v3	2
			L48s v3	1
			L64s v3	1
			L80s v3	1

## Lsv2

### Lsv2-Type1

The Lsv2-Type1 is a Dedicated Host SKU utilizing the AMD 2.55 GHz EPYC™ 7551 processor. It offers 64 physical cores, 80 vCPUs, and 640 GiB of RAM. The Lsv2-Type1 runs [Lsv2-series VMs](#).

The following packing configuration outlines the max packing of uniform VMs you can put onto a Lsv2-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
64	80	640 GiB	L8s v2	10
			L16s v2	5
			L32s v2	2
			L48s v2	1
			L64s v2	1
			L80s v2	1

## Next steps

- For more information, see the [Dedicated hosts](#) overview.
- There's a sample template, available at [Azure Quickstart Templates](#) that uses both zones and fault domains for maximum resiliency in a region.

# GPU Optimized Azure Dedicated Host SKUs

9/21/2022 • 2 minutes to read • [Edit Online](#)

Azure Dedicated Host SKUs are the combination of a VM family and a certain hardware specification. You can only deploy VMs of the VM series that the Dedicated Host SKU specifies. For example, on the Dsv3-Type3, you can only provision [Dsv3-series](#) VMs.

This document goes through the hardware specifications and VM packings for all GPU optimized Dedicated Host SKUs.

## Limitations

The sizes and hardware types available for dedicated hosts vary by region. Refer to the host [pricing page](#) to learn more.

## NVasv4

### NVasv4-Type1

The NVasv4-Type1 is a Dedicated Host SKU utilizing the AMD® Rome (EPYC™ 7V12) processor with AMD Radeon Instinct MI25 GPUs. It offers 128 physical cores, 128 vCPUs, and 448 GiB of RAM. The NVasv4-Type1 runs [NVsv4-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a NVasv4-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
128	128	448 GiB	NV4as v4	30
			NV8as v4	15
			NV16as v4	7
			NV32as v4	3

## NVsv3

### NVsv3-Type1

The NVsv3-Type1 is a Dedicated Host SKU utilizing the Intel® Broadwell (E5-2690 v4) processor with NVIDIA Tesla M60 GPUs and NVIDIA GRID technology. It offers 28 physical cores, 48 vCPUs, and 448 GiB of RAM. The NVsv3-Type1 runs [NVv3-series](#) VMs.

The following packing configuration outlines the max packing of uniform VMs you can put onto a NVsv3-Type1 host.

PHYSICAL CORES	AVAILABLE VCPUS	AVAILABLE RAM	VM SIZE	# VMS
28	48	448 GiB	NV12s v3	4

Physical Cores	Available vCPUs	Available RAM	VM Size	# VMs
			NV24s v3	2
			NV48s v3	1

## Next steps

- For more information, see the [Dedicated hosts](#) overview.
- There is sample template, available at [Azure quickstart templates](#), that uses both zones and fault domains for maximum resiliency in a region.

# Azure Dedicated Host SKU Retirement

9/21/2022 • 2 minutes to read • [Edit Online](#)

We continue to modernize and optimize Azure Dedicated Host by using the latest innovations in processor and datacenter technologies. Azure Dedicated Host is a combination of a virtual machine (VM) series and a specific Intel or AMD-based physical server. As we innovate and work with our technology partners, we also need to plan how we retire aging technology.

## Migrations required by 31 March 2023

All hardware has a finite lifespan, including the underlying hardware for Azure Dedicated Host. As we continue to modernize Azure datacenters, hardware is decommissioned and eventually retired. The hardware that runs the following Dedicated Host SKUs will be reaching end of life:

- Dsv3-Type1
- Dsv3-Type2
- Esv3-Type1
- Esv3-Type2

As a result we'll retire these Dedicated Host SKUs on 31 March 2023.

## How does the retirement of Azure Dedicated Host SKUs affect you?

The current retirement impacts the following Azure Dedicated Host SKUs:

- Dsv3-Type1
- Esv3-Type1
- Dsv3-Type2
- Esv3-Type2

Note: If you're running a Dsv3-Type3, Dsv3-Type4, an Esv3-Type3, or an Esv3-Type4 Dedicated Host, you won't be impacted.

## What actions should you take?

For manually placed VMs, you'll need to create a Dedicated Host of a newer SKU, stop the VMs on your existing Dedicated Host, reassign them to the new host, start the VMs, and delete the old host. For automatically placed VMs or for virtual machine scale sets, you'll need to create a Dedicated Host of a newer SKU, stop the VMs or virtual machine scale set, delete the old host, and then start the VMs or virtual machine scale set.

Refer to the [Azure Dedicated Host Migration Guide](#) for more detailed instructions. We recommend moving to the latest generation of Dedicated Host for your VM family.

If you have any questions, contact us through customer support.

## FAQs

### Q: Will migration result in downtime?

A: Yes, you'll need to stop/deallocate your VMs or virtual machine scale sets before moving them to the target host.

**Q: When will the other Dedicated Host SKUs retire?**

A: We'll announce Dedicated Host SKU retirements 12 months in advance of the official retirement date of a given Dedicated Host SKU.

**Q: What are the milestones for the Dsv3-Type1, Dsv3-Type2, Esv3-Type1, and Esv3-Type2 retirement?**

A:

DATE	ACTION
15 March 2022	Dsv3-Type1, Dsv3-Type2, Esv3-Type1, Esv3-Type2 retirement announcement
31 March 2023	Dsv3-Type1, Dsv3-Type2, Esv3-Type1, Esv3-Type2 retirement

**Q: What will happen to my Azure Reservation?**

A: You'll need to [exchange your reservation](#) through the Azure portal to match the new Dedicated Host SKU.

# Azure Dedicated Host SKU Retirement Migration Guide

9/21/2022 • 4 minutes to read • [Edit Online](#)

As hardware ages, it must be retired and workloads must be migrated to newer, faster, and more efficient Azure Dedicated Host SKUs. The legacy Dedicated Host SKUs should be migrated to newer Dedicated Host SKUs. The main differences between the retiring Dedicated Host SKUs and the newly recommended Dedicated Host SKUs are:

- Newer, more efficient processors
- Increased RAM
- Increased available vCPUs
- Greater regional capacity compared to the retiring Dedicated Host SKUs

Review the [FAQs](#) before you get started on migration. The next section will go over which Dedicated Host SKUs to migrate to help aid in migration planning and execution.

## Azure Dedicated Host Retirement

Some Azure Dedicated Host SKUs will be retired soon. Refer to the [Azure Dedicated Host SKU Retirement](#) documentation to learn more.

## Dsv3-Type1 and Dsv3-Type2

The Dsv3-Type1 and Dsv3-Type2 run Dsv3-series VMs, which offer a combination of vCPU, memory, and temporary storage best suited for most general-purpose workloads. We recommend migrating your existing VMs to one of the following Dedicated Host SKUs:

- Dsv3-Type3
- Dsv3-Type4

Note that both the Dsv3-Type3 and Dsv3-Type4 won't be impacted by the 31 March 2023 retirement date. We recommend moving to either the Dsv3-Type3 or Dsv3-Type4 based on regional availability, pricing, and your organization's needs.

## Esv3-Type1 and Esv3-Type2

The Esv3-Type1 and Esv3-Type2 run Esv3-series VMs, which offer a combination of vCPU, memory, and temporary storage best suited for most memory-intensive workloads. We recommend migrating your existing VMs to one of the following Dedicated Host SKUs:

- Esv3-Type3
- Esv3-Type4

Note that both the Esv3-Type3 and Esv3-Type4 won't be impacted by the 31 March 2023 retirement date. We recommend moving to either the Esv3-Type3 or Esv3-Type4 based on regional availability, pricing, and your organization's needs.

## Migration steps

To migrate your workloads to avoid Dedicated Host SKU retirement, please go through the respective steps for your manually placed VMs, automatically VMs, and virtual machine scale set on your Dedicated Host:

- [Manually Placed VMs](#)
- [Automatically Placed VMs](#)
- [VMSS](#)

1. Choose a target Dedicated Host SKU to migrate to.
2. Ensure you have quota for the VM family associated with the target Dedicated Host SKU in your given region.
3. Provision a new Dedicated Host of the target Dedicated Host SKU in the same Host Group.
4. Stop and deallocate the VM(s) on your old Dedicated Host.
5. Reassign the VM(s) to the target Dedicated Host.
6. Start the VM(s).
7. Delete the old host.

More detailed instructions can be found in the following sections.

**NOTE**

Certain sections are different for automatically placed VMs or virtual machine scale set. These differences will explicitly be called out in the respective steps.

### Ensure quota for the target VM family

Be sure that you have enough vCPU quota for the VM family of the Dedicated Host SKU that you'll be using. If you need quota, follow this guide to [request an increase in vCPU quota](#) for your target VM family in your target region. Select the Dsv3-series or Esv3-series as the VM family, depending on the target Dedicated Host SKU.

### Create a new Dedicated Host

Within the same Host Group as the existing Dedicated Host, [create a Dedicated Host](#) of the target Dedicated Host SKU.

### Stop the VM(s) or virtual machine scale set

- [PowerShell](#)
- [CLI](#)
- [Portal](#)

Refer to the PowerShell documentation to [stop a VM through PowerShell](#) or [stop a virtual machine scale set through PowerShell](#).

### Reassign the VM(s) to the target Dedicated Host

**NOTE**

Skip this step for automatically placed VMs and virtual machine scale set.

Once the target Dedicated Host has been created and the VM has been stopped, [reassign the VM to the target Dedicated Host](#).

### Start the VM(s) or virtual machine scale set

**NOTE**

Automatically placed VM(s) and virtual machine scale set require that you delete the old host *before* starting the autoplaced VM(s) or virtual machine scale set.

- [PowerShell](#)
- [CLI](#)
- [Portal](#)

Refer to the PowerShell documentation to [start a VM through PowerShell](#) or [start a virtual machine scale set through PowerShell](#).

**Delete the old Dedicated Host**

Once all VMs have been migrated from your old Dedicated Host to the target Dedicated Host, [delete the old Dedicated Host](#).

## Help and support

If you have questions, ask community experts in [Microsoft Q&A](#).

# Use Azure Spot Virtual Machines

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Using Azure Spot Virtual Machines allows you to take advantage of our unused capacity at a significant cost savings. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines. Therefore, Azure Spot Virtual Machines are great for workloads that can handle interruptions like batch processing jobs, dev/test environments, large compute workloads, and more.

The amount of available capacity can vary based on size, region, time of day, and more. When deploying Azure Spot Virtual Machines, Azure will allocate the VMs if there's capacity available, but there's no SLA for these VMs. An Azure Spot Virtual Machine offers no high availability guarantees. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines with 30-seconds notice.

## Eviction policy

VMs can be evicted based on capacity or the max price you set. When creating an Azure Spot Virtual Machine, you can set the eviction policy to *Deallocate* (default) or *Delete*.

The *Deallocate* policy moves your VM to the stopped-deallocated state, allowing you to redeploy it later. However, there's no guarantee that the allocation will succeed. The deallocated VMs will count against your quota and you'll be charged storage costs for the underlying disks.

If you would like your VM to be deleted when it's evicted, you can set the eviction policy to *delete*. The evicted VMs are deleted together with their underlying disks, so you'll not continue to be charged for the storage.

You can opt in to receive in-VM notifications through [Azure Scheduled Events](#). This will notify you if your VMs are being evicted and you will have 30 seconds to finish any jobs and perform shutdown tasks prior to the eviction.

OPTION	OUTCOME
Max price is set to $\geq$ the current price.	VM is deployed if capacity and quota are available.
Max price is set to $<$ the current price.	The VM isn't deployed. You'll get an error message that the max price needs to be $\geq$ current price.
Restarting a stopped/deallocated VM if the max price is $\geq$ the current price	If there's capacity and quota, then the VM is deployed.
Restarting a stopped/deallocated VM if the max price is $<$ the current price	You'll get an error message that the max price needs to be $\geq$ current price.
Price for the VM has gone up and is now $>$ the max price.	The VM gets evicted. You get a 30s notification before actual eviction.
After eviction, the price for the VM goes back to being $<$ the max price.	The VM won't be automatically restarted. You can restart the VM yourself, and it will be charged at the current price.

OPTION	OUTCOME
If the max price is set to <code>-1</code>	The VM won't be evicted for pricing reasons. The max price will be the current price, up to the price for standard VMs. You'll never be charged above the standard price.
Changing the max price	You need to deallocate the VM to change the max price. Deallocate the VM, set a new max price, then update the VM.

#### TIP

Check out our [Azure Virtual Machine Spot Eviction](#) guide to learn how to create a reliable interruptible workload in Azure.

## Limitations

The following VM sizes aren't supported for Azure Spot Virtual Machines:

- B-series
- Promo versions of any size (like Dv2, NV, NC, H promo sizes)

Azure Spot Virtual Machines can be deployed to any region, except Microsoft Azure China 21Vianet.

The following [offer types](#) are currently supported:

- Enterprise Agreement
- Pay-as-you-go offer code (003P)
- Sponsored (0036P and 0136P)
- For Cloud Service Provider (CSP), see the [Partner Center](#) or contact your partner directly.

## Pricing

Pricing for Azure Spot Virtual Machines is variable, based on region and SKU. For more information, see VM pricing for [Linux](#) and [Windows](#).

You can also query pricing information using the [Azure retail prices API](#) to query for information about Spot pricing. The `meterName` and `skuName` will both contain `spot`.

With variable pricing, you have option to set a max price, in US dollars (USD), using up to five decimal places. For example, the value `0.98765` would be a max price of \$0.98765 USD per hour. If you set the max price to be `-1`, the VM won't be evicted based on price. The price for the VM will be the current price for spot or the price for a standard VM, whichever is less, as long as there's capacity and quota available.

## Pricing and eviction history

### Portal

You can see historical pricing and eviction rates per size in a region in the portal. Select **View pricing history and compare prices in nearby regions** to see a table or graph of pricing for a specific size. The pricing and eviction rates in the following images are only examples.

### Chart:

## Set maximum price for Azure Spot

X

The maximum price you input will be used to determine when your VM is allocated. If the platform cost is less than or equal to your maximum price, your VM will be allocated. All prices are submitted in USD (\$). [Learn more about maximum prices for Azure Spot VMs.](#)

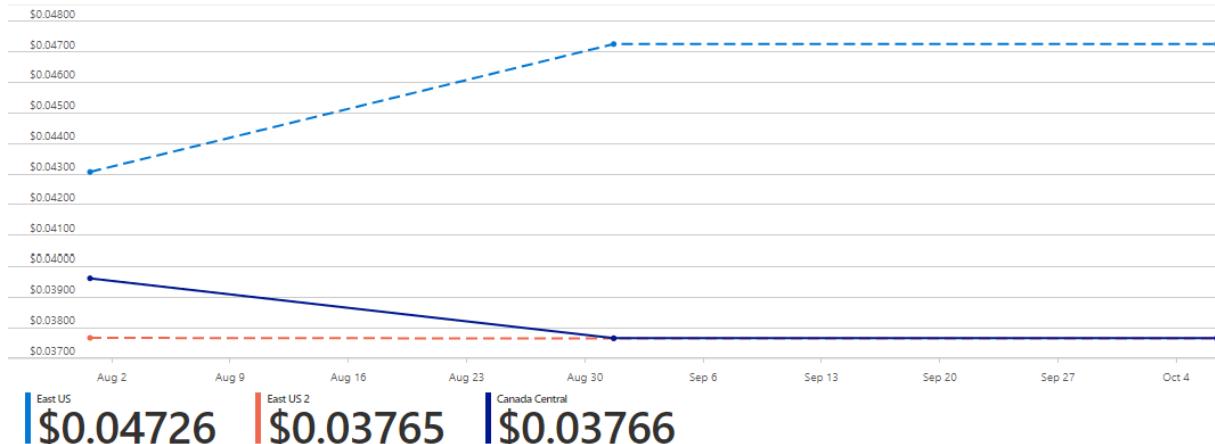
Pricing history for

Standard\_DS3\_v2 - 4 vcpus, 14 GiB memory

Time span : **Three months**

Currency : **USD**

≡ Switch to table



**East US**  
**\$0.04726**

Eviction rate: 0-5%  
Select

**East US 2**  
**\$0.03765**

Eviction rate: 5-10%  
Select

**Canada Central**  
**\$0.03766**

Eviction rate: 0-5%  
Selected

Maximum price you want to pay per hour (USD) ⓘ

0.293

OK

Table:

## Set maximum price for Azure Spot

X

The maximum price you input will be used to determine when your VM is allocated. If the platform cost is less than or equal to your maximum price, your VM will be allocated. All prices are submitted in USD (\$). [Learn more about maximum prices for Azure Spot VMs.](#)

Pricing history for

Standard\_DS3\_v2 - 4 vcpus, 14 GiB memory

Time span : **Three months**

Currency : **USD**

≡ Switch to chart

Date	East US	East US 2	Canada Central
7/31/2020	\$0.04307	\$0.03767	\$0.03962
8/31/2020	\$0.04726	\$0.03765	\$0.03766
10/6/2020	\$0.04726	\$0.03765	\$0.03766

**East US**  
**\$0.04726**

Eviction rate: 0-5%  
Select

**East US 2**  
**\$0.03765**

Eviction rate: 5-10%  
Select

**Canada Central**  
**\$0.03766**

Eviction rate: 0-5%  
Selected

Maximum price you want to pay per hour (USD) ⓘ

0.293

OK

Azure Resource Graph

You can programmatically access relevant Spot VM SKU data through [Azure Resource Graph](#). Get pricing history in the last 90 days and eviction rates for the last 28 trailing days to identify SKUs that better meet your specific

needs.

Key benefits:

- Query Spot eviction rates and the last few months of Spot prices programmatically through ARM or the [ARG Explorer in Azure portal](#)
- Create a custom query to extract the specific data relevant to your scenario with the ability to filter across a variety of parameters, such as SKU and region
- Easily compare data across multiple regions and SKUs
- Find a different Spot SKU or region with a lower price and/or eviction rate

Try out the following sample queries for Spot pricing history and eviction rates using the [ARG Explorer in Azure portal](#). Spot pricing history and eviction rates data are available in the `SpotResources` table.

**Spot pricing history sample query:**

```
SpotResources
|wheretype=~'microsoft.compute/skupsotpricehistory/ostype/location'
|wheresku.namein~('standard_d2s_v4','standard_d4s_v4')
|whereproperties.osType=~'linux'
|wherelocationin~('eastus','southcentralus')
|projectskuName=tostring(sku.name),osType=tostring(properties.osType),location,latestSpotPriceUSD=todouble(properties.spotPrices[0].priceUSD)
|orderbylatestSpotPriceUSDAsc
```

**Spot eviction rates sample query:**

```
SpotResources
|wheretype=~'microsoft.compute/skuspotevictionrate/location'
|wheresku.namein~('standard_d2s_v4','standard_d4s_v4')
|wherelocationin~('eastus','southcentralus')
|projectskuName=tostring(sku.name),location,spotEvictionRate=tostring(properties.evictionRate)
|orderbyskuNameasc,locationasc
```

Alternatively, try out the [ARG REST API](#) to get the pricing history and eviction rate history data.

## Frequently asked questions

**Q:** Once created, is an Azure Spot Virtual Machine the same as regular standard VM?

**A:** Yes, except there's no SLA for Azure Spot Virtual Machines and they can be evicted at any time.

**Q:** What to do when you get evicted, but still need capacity?

**A:** We recommend you use standard VMs instead of Azure Spot Virtual Machines if you need capacity right away.

**Q:** How is quota managed for Azure Spot Virtual Machines?

**A:** Azure Spot Virtual Machines will have a separate quota pool. Spot quota will be shared between VMs and scale-set instances. For more information, see [Azure subscription and service limits, quotas, and constraints](#).

**Q:** Can I request for additional quota for Azure Spot Virtual Machines?

**A:** Yes, you'll be able to submit the request to increase your quota for Azure Spot Virtual Machines through the [standard quota request process](#).

**Q:** Where can I post questions?

A: You can post and tag your question with `azure-spot` at [Q&A](#).

Q: How can I change the max price for a spot VM?

A: Before you can change the max price, you need to deallocate the VM. Then you can change the max price in the portal, from the **Configuration** section for the VM.

## Next steps

Use the [CLI](#), [portal](#), [ARM template](#), or [PowerShell](#) to deploy Azure Spot Virtual Machines.

You can also deploy a [scale set with Azure Spot Virtual Machine instances](#).

If you encounter an error, see [Error codes](#).

# Deploy Azure Spot Virtual Machines using the Azure CLI

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Using [Azure Spot Virtual Machines](#) allows you to take advantage of our unused capacity at a significant cost savings. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines. Therefore, Azure Spot Virtual Machines are great for workloads that can handle interruptions like batch processing jobs, dev/test environments, large compute workloads, and more.

Pricing for Azure Spot Virtual Machines is variable, based on region and SKU. For more information, see VM pricing for [Linux](#) and [Windows](#).

You have option to set a max price you are willing to pay, per hour, for the VM. The max price for an Azure Spot Virtual Machine can be set in US dollars (USD), using up to 5 decimal places. For example, the value `0.98765` would be a max price of \$0.98765 USD per hour. If you set the max price to be `-1`, the VM won't be evicted based on price. The price for the VM will be the current price for Azure Spot Virtual Machine or the price for a standard VM, whichever is less, as long as there is capacity and quota available. For more information about setting the max price, see [Azure Spot Virtual Machines - Pricing](#).

The process to create an Azure Spot Virtual Machine using the Azure CLI is the same as detailed in the [quickstart article](#). Just add the '--priority Spot' parameter, set the `--eviction-policy` to either Deallocate (this is the default) or `Delete`, and provide a max price or `-1`.

## Install Azure CLI

To create Azure Spot Virtual Machines, you need to be running the Azure CLI version 2.0.74 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install the Azure CLI](#).

Sign in to Azure using [az login](#).

```
az login
```

## Create an Azure Spot Virtual Machine

This example shows how to deploy a Linux Azure Spot Virtual Machine that will not be evicted based on price. The eviction policy is set to deallocate the VM, so that it can be restarted at a later time. If you want to delete the VM and the underlying disk when the VM is evicted, set `--eviction-policy` to `Delete`.

```
az group create -n mySpotGroup -l eastus
az vm create \
    --resource-group mySpotGroup \
    --name myVM \
    --image UbuntuLTS \
    --admin-username azureuser \
    --generate-ssh-keys \
    --priority Spot \
    --max-price -1 \
    --eviction-policy Deallocate
```

After the VM is created, you can query to see the max billing price for all of the VMs in the resource group.

```
az vm list \
-g mySpotGroup \
--query '[].{Name:name, MaxPrice:billingProfile.maxPrice}' \
--output table
```

## Simulate an eviction

You can simulate an eviction of an Azure Spot Virtual Machine using REST, PowerShell, or the CLI, to test how well your application will respond to a sudden eviction.

In most cases, you will want to use the REST API [Virtual Machines - Simulate Eviction](#) to help with automated testing of applications. For REST, a `Response Code: 204` means the simulated eviction was successful. You can combine simulated evictions with the [Scheduled Event service](#), to automate how your app will respond when the VM is evicted.

To see scheduled events in action, watch [Azure Friday - Using Azure Scheduled Events to prepare for VM maintenance](#).

### Quick test

For a quick test to show how a simulated eviction will work, let's walk through querying the scheduled event service to see what it looks like when you simulate an eviction using the Azure CLI.

The Scheduled Event service is enabled for your service the first time you make a request for events.

Remote into your VM, and then open a command prompt.

From the command prompt on your VM, type:

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2019-08-01
```

This first response could take up to 2 minutes. From now on, they should display output almost immediately.

From a computer that has the Azure CLI installed (like your local machine), simulate an eviction using `az vm simulate-eviction`. Replace the resource group name and VM name with your own.

```
az vm simulate-eviction --resource-group mySpotRG --name mySpot
```

The response output will have `Status: Succeeded` if the request was successfully made.

Quickly go back to your remote connection to your Spot Virtual Machine and query the Scheduled Events endpoint again. Repeat the following command until you get an output that contains more information:

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2019-08-01
```

When the Scheduled Event Service gets the eviction notification, you will get a response that looks similar to this:

```
{"DocumentIncarnation":1,"Events":[{"EventId":"A123BC45-1234-5678-AB90-ABCDEF123456","EventStatus":"Scheduled","EventType":"Preempt","ResourceType":"VirtualMachine","Resources":["myspotvm"],"NotBefore":"Tue, 16 Mar 2021 00:58:46 GMT","Description":"","EventSource":"Platform"}]}
```

You can see that `"EventType": "Preempt"`, and the resource is the VM resource `"Resources": ["myspotvm"]`.

You can also see when the VM will be evicted by checking the `"NotBefore"` - the VM will not be evicted before the time given, so that is your window for your application to gracefully close out.

## Next steps

You can also create an Azure Spot Virtual Machine using [Azure PowerShell](#), [portal](#), or a [template](#).

Query current pricing information using the [Azure retail prices API](#) for information about Azure Spot Virtual Machine. The `meterName` and `skuName` will both contain `Spot`.

If you encounter an error, see [Error codes](#).

# Deploy Azure Spot Virtual Machines using the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale sets

Using [Azure Spot Virtual Machines](#) allows you to take advantage of our unused capacity at a significant cost savings. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines. Therefore, Azure Spot Virtual Machines are great for workloads that can handle interruptions like batch processing jobs, dev/test environments, large compute workloads, and more.

Pricing for Azure Spot Virtual Machines is variable, based on region and SKU. For more information, see VM pricing for [Linux](#) and [Windows](#). For more information about setting the max price, see [Azure Spot Virtual Machines - Pricing](#).

You have option to set a max price you are willing to pay, per hour, for the VM. The max price for an Azure Spot Virtual Machine can be set in US dollars (USD), using up to 5 decimal places. For example, the value `0.05701` would be a max price of \$0.05701 USD per hour. If you set the max price to be `-1`, the VM won't be evicted based on price. The price for the VM will be the current price for spot or the price for a standard VM, whichever is less, as long as there is capacity and quota available.

When the VM is evicted, you have the option to either delete the VM and the underlying disk or deallocate the VM so that it can be restarted later.

## Create the VM

When you are deploying a VM, you can choose to use an Azure spot instance.

On the **Basics** tab, in the **Instance details** section, **No** is the default for using an Azure spot instance.

The screenshot shows the 'Instance details' section of the Azure portal's VM creation form. It includes fields for Virtual machine name, Region, Availability options, Image, and Azure Spot instance selection. The 'Azure Spot instance' field is highlighted with a red border, and the 'No' radio button is selected.

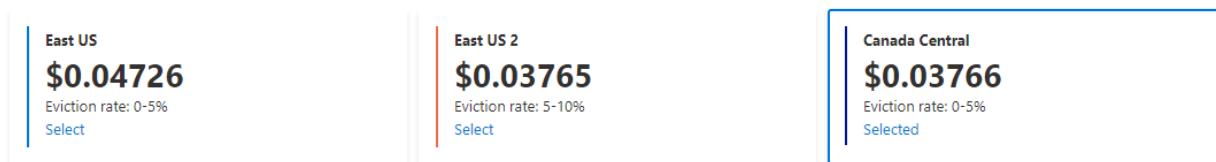
Instance details	
Virtual machine name *	<input type="text"/>
Region *	(US) West US
Availability options	No infrastructure redundancy required
Image *	Ubuntu Server 18.04 LTS <a href="#">Browse all public and private images</a>
Azure Spot instance	<input type="radio"/> Yes <input checked="" type="radio"/> No
Size *	<a href="#">Select size</a>

If you select **Yes**, the section expands and you can choose your [eviction type and eviction policy](#).

Azure Spot instance <a href="#">(i)</a>	<input checked="" type="radio"/> Yes <input type="radio"/> No
Eviction type <a href="#">(i)</a>	<input checked="" type="radio"/> Capacity only: evict virtual machine when Azure needs the capacity for pay as you go workloads. Your max price is set to the pay as you go rate. <input type="radio"/> Price or capacity: choose a max price and Azure will evict your virtual machine when the cost of the instance is greater than your max price or when Azure needs the capacity for pay as you go workloads.
Eviction policy <a href="#">(i)</a>	<input checked="" type="radio"/> Stop / Deallocate <input type="radio"/> Delete

You can also compare the pricing and eviction rates with other similar regions by selecting [View pricing history and compare prices in nearby regions](#).

In this example, the Canada Central region is less expensive and has a lower eviction rate than the East US region.



You can change the region by selecting the choice that works best for you and then selecting OK.

## Simulate an eviction

You can [simulate an eviction](#) of an Azure Spot Virtual Machine, to test how well your application will respond to a sudden eviction.

Replace the following with your information:

- `subscriptionId`
- `resourceGroupName`
- `vmName`

```
POST
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}/simulateEviction?api-version=2020-06-01
```

Response Code: 204 means the simulated eviction was successful.

## Next steps

You can also create Azure Spot Virtual Machines using [PowerShell](#), [CLI](#), or a [template](#).

# Deploy Azure Spot Virtual Machines using Azure PowerShell

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

Using [Azure Spot Virtual Machines](#) allows you to take advantage of our unused capacity at a significant cost savings. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines. Therefore, Azure Spot Virtual Machines are great for workloads that can handle interruptions like batch processing jobs, dev/test environments, large compute workloads, and more.

Pricing for Azure Spot Virtual Machines is variable, based on region and SKU. For more information, see VM pricing for [Linux](#) and [Windows](#). For more information about setting the max price, see [Azure Spot Virtual Machines - Pricing](#).

You have option to set a max price you are willing to pay, per hour, for the VM. The max price for an Azure Spot Virtual Machine can be set in US dollars (USD), using up to 5 decimal places. For example, the value `0.98765` would be a max price of \$0.98765 USD per hour. If you set the max price to be `-1`, the VM won't be evicted based on price. The price for the VM will be the current price for spot or the price for a standard VM, whichever is less, as long as there is capacity and quota available.

## Create the VM

Create a spotVM using [New-AzVmConfig](#) to create the configuration. Include `-Priority Spot` and set `-MaxPrice` to either:

- `-1` so the VM is not evicted based on price.
- a dollar amount, up to 5 digits. For example, `-MaxPrice .98765` means that the VM will be deallocated once the price for a spotVM goes about \$.98765 per hour.

This example creates a spotVM that will not be deallocated based on pricing (only when Azure needs the capacity back). The eviction policy is set to deallocate the VM, so that it can be restarted at a later time. If you want to delete the VM and the underlying disk when the VM is evicted, set `-EvictionPolicy` to `Delete` in [New-AzVMConfig](#).

```

$resourceGroup = "mySpotRG"
$location = "eastus"
$vmName = "mySpotVM"
$cred = Get-Credential -Message "Enter a username and password for the virtual machine."
New-AzResourceGroup -Name $resourceGroup -Location $location
$subnetConfig = New-AzVirtualNetworkSubnetConfig ` 
    -Name mySubnet -AddressPrefix 192.168.1.0/24
$vnet = New-AzVirtualNetwork -ResourceGroupName $resourceGroup ` 
    -Location $location -Name MYvNET -AddressPrefix 192.168.0.0/16 ` 
    -Subnet $subnetConfig
$pip = New-AzPublicIpAddress -ResourceGroupName $resourceGroup -Location $location ` 
    -Name "mypublicdns$(Get-Random)" -AllocationMethod Static -IdleTimeoutInMinutes 4
$nsgRuleRDP = New-AzNetworkSecurityRuleConfig -Name myNetworkSecurityGroupRuleRDP -Protocol Tcp ` 
    -Direction Inbound -Priority 1000 -SourceAddressPrefix * -SourcePortRange * -DestinationAddressPrefix * ` 
    -DestinationPortRange 3389 -Access Deny
$nsg = New-AzNetworkSecurityGroup -ResourceGroupName $resourceGroup -Location $location ` 
    -Name myNetworkSecurityGroup -SecurityRules $nsgRuleRDP
$nic = New-AzNetworkInterface -Name myNic -ResourceGroupName $resourceGroup -Location $location ` 
    -SubnetId $vnet.Subnets[0].Id -PublicIpAddressId $pip.Id -NetworkSecurityGroupId $nsg.Id

# Create a virtual machine configuration and set this to be an Azure Spot Virtual Machine

$vmConfig = New-AzVMConfig -VMName $vmName -VMSize Standard_D1 -Priority "Spot" -MaxPrice -1 -EvictionPolicy
Deallocate | ` 
Set-AzVMOperatingSystem -Windows -ComputerName $vmName -Credential $cred | ` 
Set-AzVMSourceImage -PublisherName MicrosoftWindowsServer -Offer WindowsServer -Skus 2016-Datacenter - 
Version latest | ` 
Add-AzVMNetworkInterface -Id $nic.Id

New-AzVM -ResourceGroupName $resourceGroup -Location $location -VM $vmConfig

```

After the VM is created, you can query to see the max price for all of the VMs in the resource group.

```

Get-AzVM -ResourceGroupName $resourceGroup | ` 
Select-Object Name,@{Name="maxPrice"; Expression={$_['BillingProfile.MaxPrice}}}

```

## Simulate an eviction

You can simulate an eviction of an Azure Spot Virtual Machine using REST, PowerShell, or the CLI, to test how well your application will respond to a sudden eviction.

In most cases, you will want to use the REST API [Virtual Machines - Simulate Eviction](#) to help with automated testing of applications. For REST, a `Response Code: 204` means the simulated eviction was successful. You can combine simulated evictions with the [Scheduled Event service](#), to automate how your app will respond when the VM is evicted.

To see scheduled events in action, watch Azure Friday - Using Azure Scheduled Events to prepare for VM maintenance.

### Quick test

For a quick test to show how a simulated eviction will work, let's walk through querying the scheduled event service to see what it looks like when you simulate an eviction using PowerShell.

The Scheduled Event service is enabled for your service the first time you make a request for events.

Remote into your VM, and then open a command prompt.

From the command prompt on your VM, type:

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2019-08-01
```

This first response could take up to 2 minutes. From now on, they should display output almost immediately.

From a computer that has the Az PowerShell module installed (like your local machine), simulate an eviction using [Set-AzVM](#). Replace the resource group name and VM name with your own.

```
Set-AzVM -ResourceGroupName "mySpotRG" -Name "mySpotVM" -SimulateEviction
```

The response output will have `Status: Succeeded` if the request was successfully made.

Quickly go back to your remote connection to your Spot Virtual Machine and query the Scheduled Events endpoint again. Repeat the following command until you get an output that contains more information:

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2019-08-01
```

When the Scheduled Event Service gets the eviction notification, you will get a response that looks similar to this:

```
{"DocumentIncarnation":1,"Events":[{"EventId":"A123BC45-1234-5678-AB90-ABCDEF123456","EventStatus":"Scheduled","EventType":"Preempt","ResourceType":"VirtualMachine","Resources":["myspotvm"],"NotBefore":"Tue, 16 Mar 2021 00:58:46 GMT","Description":"","EventSource":"Platform"}]}
```

You can see that `"EventType": "Preempt"`, and the resource is the VM resource `"Resources": ["myspotvm"]`.

You can also see when the VM will be evicted by checking the `"NotBefore"` value. The VM will not be evicted before the time given in `NotBefore`, so that is your window for your application to gracefully close out.

## Next steps

You can also create an Azure Spot Virtual Machine using the [Azure CLI](#), [portal](#) or a [template](#).

Query current pricing information using the [Azure retail prices API](#) for information about Azure Spot Virtual Machine pricing. The `meterName` and `skuName` will both contain `Spot`.

If you encounter an error, see [Error codes](#).

# Deploy Azure Spot Virtual Machines using a Resource Manager template

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Using [Azure Spot Virtual Machines](#) allows you to take advantage of our unused capacity at a significant cost savings. At any point in time when Azure needs the capacity back, the Azure infrastructure will evict Azure Spot Virtual Machines. Therefore, Azure Spot Virtual Machines are great for workloads that can handle interruptions like batch processing jobs, dev/test environments, large compute workloads, and more.

Pricing for Azure Spot Virtual Machines is variable, based on region and SKU. For more information, see VM pricing for [Linux](#) and [Windows](#).

You have option to set a max price you are willing to pay, per hour, for the VM. The max price for an Azure Spot Virtual Machine can be set in US dollars (USD), using up to 5 decimal places. For example, the value `0.98765` would be a max price of \$0.98765 USD per hour. If you set the max price to be `-1`, the VM won't be evicted based on price. The price for the VM will be the current price for Azure Spot Virtual Machines or the price for a standard VM, whichever is less, as long as there is capacity and quota available. For more information about setting the max price, see [Azure Spot Virtual Machines - Pricing](#).

## Use a template

For Azure Spot Virtual Machine template deployments, use `"apiVersion": "2019-03-01"` or later. Add the `priority`, `evictionPolicy` and `billingProfile` properties to in your template:

```
"priority": "Spot",
"evictionPolicy": "Deallocate",
"billingProfile": {
    "maxPrice": -1
}
```

Here is a sample template with the added properties for an Azure Spot Virtual Machine. Replace the resource names with your own and `<password>` with a password for the local administrator account on the VM.

```
{
    "$schema": "http://schema.management.azure.com/schemas/2019-03-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {},
    "variables": {
        "vnetId": "/subscriptions/ec9fcd04-e188-48b9-abfc-
abcd515f1836/resourceGroups/spotVM/providers/Microsoft.Network/virtualNetworks/spotVM",
        "subnetName": "default",
        "networkInterfaceName": "spotVMNIC",
        "publicIpAddressName": "spotVM-ip",
        "publicIpAddressType": "Dynamic",
        "publicIpAddressSku": "Basic",
        "virtualMachineName": "spotVM",
        "osDiskType": "Premium_LRS",
        "virtualMachineSize": "Standard_D2s_v3",
        "adminUsername": "azureuser",
        "adminPassword": "<password>",
        "diagnosticsStorageAccountName": "diagstoragespot2019",
```

```

    "diagnosticsStorageAccountId": "Microsoft.Storage/storageAccounts/diagstoragespot2019",
    "diagnosticsStorageAccountType": "Standard_LRS",
    "diagnosticsStorageAccountKind": "Storage",
    "subnetRef": "[concat(variables('vnetId'), '/subnets/', variables('subnetName'))]"
},
"resources": [
{
    "name": "spotVM",
    "type": "Microsoft.Network/networkInterfaces",
    "apiVersion": "2019-03-01",
    "location": "eastus",
    "dependsOn": [
        "[concat('Microsoft.Network/publicIpAddresses/', variables('publicIpAddressName'))]"
    ],
    "properties": {
        "ipConfigurations": [
            {
                "name": "ipconfig1",
                "properties": {
                    "subnet": {
                        "id": "[variables('subnetRef')]"
                    },
                    "privateIPAllocationMethod": "Dynamic",
                    "publicIpAddress": {
                        "id": "[resourceId(resourceGroup().name,
'Microsoft.Network/publicIpAddresses', variables('publicIpAddressName'))]"
                    }
                }
            }
        ]
    }
},
{
    "name": "[variables('publicIpAddressName')]",
    "type": "Microsoft.Network/publicIpAddresses",
    "apiVersion": "2019-02-01",
    "location": "eastus",
    "properties": {
        "publicIpAllocationMethod": "[variables('publicIpAddressType')]"
    },
    "sku": {
        "name": "[variables('publicIpAddressSku')]"
    }
},
{
    "name": "[variables('virtualMachineName')]",
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2019-03-01",
    "location": "eastus",
    "dependsOn": [
        "[concat('Microsoft.Network/networkInterfaces/', variables('networkInterfaceName'))]",
        "[concat('Microsoft.Storage/storageAccounts/', variables('diagnosticsStorageAccountName'))]"
    ],
    "properties": {
        "hardwareProfile": {
            "vmSize": "[variables('virtualMachineSize')]"
        },
        "storageProfile": {
            "osDisk": {
                "createOption": "fromImage",
                "managedDisk": {
                    "storageAccountType": "[variables('osDiskType')]"
                }
            },
            "imageReference": {
                "publisher": "Canonical",
                "offer": "UbuntuServer",
                "sku": "18.04-LTS",
                "version": "latest"
            }
        }
    }
}
]
}

```

```

        }
    },
    "networkProfile": {
        "networkInterfaces": [
            {
                "id": "[resourceId('Microsoft.Network/networkInterfaces',
variables('networkInterfaceName'))]"
            }
        ]
    },
    "osProfile": {
        "computerName": "[variables('virtualMachineName')]",
        "adminUsername": "[variables('adminUsername')]",
        "adminPassword": "[variables('adminPassword')]"
    },
    "diagnosticsProfile": {
        "bootDiagnostics": {
            "enabled": true,
            "storageUri": "[concat('https://', variables('diagnosticsStorageAccountName'),
'.blob.core.windows.net/'])"
        }
    },
    "priority": "Spot",
    "evictionPolicy": "Deallocate",
    "billingProfile": {
        "maxPrice": -1
    }
}
},
{
    "name": "[variables('diagnosticsStorageAccountName')]",
    "type": "Microsoft.Storage/storageAccounts",
    "apiVersion": "2019-04-01",
    "location": "eastus",
    "properties": {},
    "kind": "[variables('diagnosticsStorageAccountKind')]",
    "sku": {
        "name": "[variables('diagnosticsStorageAccountType')]"
    }
}
],
"outputs": {
    "adminUsername": {
        "type": "string",
        "value": "[variables('adminUsername')]"
    }
}
}

```

## Simulate an eviction

You can [simulate an eviction](#) of an Azure Spot Virtual Machine, to testing how well your application will respond to a sudden eviction.

Replace the following with your information:

- 
- 
- 

```

POST
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}/simulateEviction?api-version=2020-06-01

```

## Next steps

You can also create an Azure Spot Virtual Machine using [Azure PowerShell](#) or the [Azure CLI](#).

Query current pricing information using the [Azure retail prices API](#) for information about Azure Spot Virtual Machine pricing. The `meterName` and `skuName` will both contain `Spot`.

If you encounter an error, see [Error codes](#).

# Spot Virtual Machine size recommendation

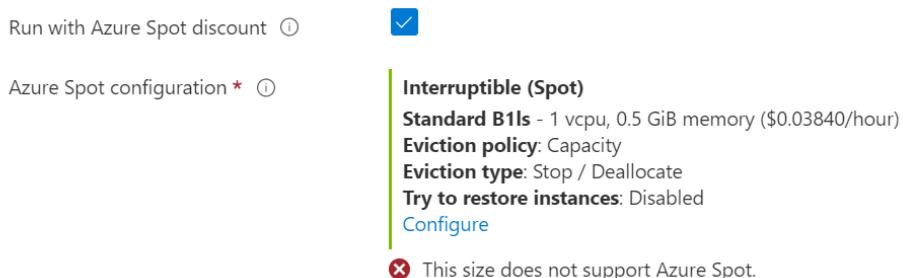
9/21/2022 • 2 minutes to read • [Edit Online](#)

The Spot VM size recommendations tool is an easy way to view and select alternative VM sizes that are better suited for your stateless, flexible, and fault tolerant workload needs during the Virtual Machine Scale Set deployment process in the Azure portal. This tool allows Azure to recommend appropriate VM sizes to you after you filter by region, price, and eviction rate. You can further filter the recommended VMs list by size, type, generation, and disk (premium or ephemeral OS disk).

## Azure portal

You can access Azure's size recommendations through the virtual machine scale sets creation process in the Azure portal. The following steps will instruct you on how to access this tool during that process.

1. Log in to the [Azure portal](#).
2. In the search bar, search for and select **Virtual machine scale sets**.
3. Select **Create** on the **Virtual machine scale sets** page.
4. In the **Basics** tab, fill out the required fields.
5. Under **Instance details**, select **Run with Azure Spot discount**.



6. In the same section, under **Azure Spot configuration**, select **Configure**.
7. On the **Azure Spot configuration** page, in the **Spot details** tab, go to the **Size** selector.
8. Expand the **Size** drop-down and select **See all sizes** option at the bottom of the list.

## Azure Spot configuration

X

### Spot details !

Size recommendations

Azure Spot offers unused Azure capacity at a discounted rate. Your workloads should be able to tolerate interruptions or infrastructure loss when Azure needs the capacity elsewhere. [Learn more about Azure Spot instances](#)

Eviction type (i)
 Capacity only
 Standard\_B1ls - 1 vcpu, 0.5 GiB memory (Price unavailable) (i)

Standard\_D2s\_v3 - 2 vcpus, 2 GiB memory (\$0.03840/hour)

## Your recently used sizes

Standard\_B1ls - 1 vcpu, 0.5 GiB memory (Price unavailable) (i)

Standard\_D2s\_v3 - 2 vcpus, 2 GiB memory (\$0.03840/hour)

## Sizes recommended by image publisher

Standard\_D2s\_v3 - 2 vcpus, 2 GiB memory (\$0.03840/hour)

Standard\_D4s\_v3 - 4 vcpus, 4 GiB memory (\$0.07680/hour)

Standard\_E2s\_v3 - 2 vcpus, 2 GiB memory (\$0.05040/hour)

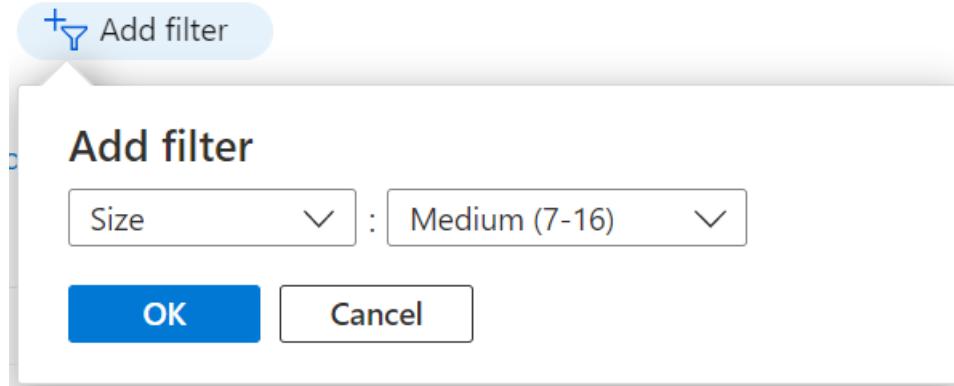
See all sizes

Standard\_B1ls - 1 vcpu, 0.5 GiB memory (Price unavailable) (i)Eviction policy (i)Try to restore instances (i)Maximum price you want to pay per hour  
(USD) \*Size (i)

This size does not support Azure Spot.

9. On the **Select a VM size** page, click **Add filter**.

10. You can choose which filters to apply. For this example, we will only apply **Size** and set it to *Medium (7-16)* for the number of vCPU.



11. Click **OK**.

12. From the resulting list of VMs, select a preferred VM size.

13. Click **Select** at the bottom to continue.

14. Back on the **Spot details** tab, click **Next** to go to the next tab.

15. The **Size recommendations** tab allows you to view and select alternative VM sizes that are better suited for your stateless, flexible, and fault tolerant workload needs with regard to region, pricing, and eviction rates.

## Azure Spot configuration

[Spot details](#) [Size recommendations](#)

Tell us what size you're interested in and we'll show you similar sizes across selected regions that might be better suited for your needs.  
Show recommendations similar to [\(i\)](#)

Standard\_D8s\_v3 - 8 vcpus, 8 GiB memory (\$0.15360/hour) [▼](#) Display price : Hourly Region : 3 selected Maximum price per hour : \$10.00000 Eviction rate : <5% - >20%

SKU	vCPUs	RAM (GiB)	Region	Savings	Price history over 90 days	Eviction rate	Cost per hour
F8s_v2	8	16	East US	87%	 \$0.01274	<5%	\$0.04475
D8as_v4	8	32	East US	90%	 \$0.03118	<5%	\$0.03994
E8as_v4	8	64	East US	90%	 \$0.04531	<5%	\$0.05242
D8s_v3	8	32	East US	60%	 \$0.00491	<5%	\$0.15360
DS12_v2	4	28	East US	60%	 \$0.01862	<5%	\$0.14833
D8ds_v4	8	32	East US	60%	 \$0.01338	<5%	\$0.18071
E8s_v3	8	64	East US	60%	 \$0.00383	<5%	\$0.20160
E8ds_v4	8	64	East US	60%	 \$0.00772	<5%	\$0.23029
D8s_v4	8	32	East US 2	60%	 \$0.01463	5% - 10%	\$0.15352
D8as_v4	8	32	East US 2	90%	 \$0.01321	<5%	\$0.03840

[Save](#)[< Previous](#)[Next >](#)

Actual discounts may vary based on region, VM type, and Azure compute capacity available when the workload is deployed.

16. Make your selection and click **Save**.

17. Continue through the virtual machine scale set creation process.

## Next steps

[Learn more about Spot virtual machines](#)

# Error messages for Azure Spot Virtual Machines and scale sets

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Here are some possible error codes you could receive when using Azure Spot Virtual Machines and scale sets.

KEY	MESSAGE	DESCRIPTION
SkuNotAvailable	The requested tier for resource '<resource>' is currently not available in location '<location>' for subscription '<subscriptionID>'. Try another tier or deploy to a different location.	There is not enough Azure Spot Virtual Machine capacity in this location to create your VM or scale set instance.
EvictionPolicyCanBeSetOnlyOnAzureSpotVirtualMachines	Eviction policy can be set only on Azure Spot Virtual Machines.	This VM is not an Azure Spot Virtual Machine, so you can't set the eviction policy.
AzureSpotVMNotSupportedInAvailabilitySet	Azure Spot Virtual Machine is not supported in Availability Set.	You need to choose to either use an Azure Spot Virtual Machine or use a VM in an availability set, you can't choose both.
AzureSpotFeatureNotEnabledForSubscription	Subscription not enabled with Azure Spot Virtual Machine feature.	Use a subscription that supports Azure Spot Virtual Machines.
VMPriorityCannotBeApplied	The specified priority value '{0}' cannot be applied to the Virtual Machine '{1}' since no priority was specified when the Virtual Machine was created.	Specify the priority when the VM is created.
SpotPriceGreater ThanProvidedMaxPrice	Unable to perform operation '{0}' since the provided max price '{1} USD' is lower than the current spot price '{2} USD' for Azure Spot Virtual Machine size '{3}'.	Select a higher max price. For more information, see pricing information for <a href="#">Linux</a> or <a href="#">Windows</a> .
MaxPriceValueInvalid	Invalid max price value. The only supported values for max price are -1 or a decimal greater than zero. Max price value of -1 indicates the Azure Spot Virtual Machine will not be evicted for price reasons.	Enter a valid max price. For more information, see pricing for <a href="#">Linux</a> or <a href="#">Windows</a> .
MaxPriceChangeNotAllowedForAllocatedVMs	Max price change is not allowed when the VM '{0}' is currently allocated. Deallocate and try again.	Stop\Deallocate the VM so that you can change the max price.
MaxPriceChangeNotAllowed	Max price change is not allowed.	You cannot change the max price for this VM.

KEY	MESSAGE	DESCRIPTION
AzureSpotIsNotSupportedForThisAPIVersion	Azure Spot Virtual Machine is not supported for this API version.	The API version needs to be 2019-03-01.
AzureSpotIsNotSupportedForThisVMSize	Azure Spot Virtual Machine is not supported for this VM size {0}.	Select another VM size. For more information, see <a href="#">Azure Spot Virtual Machines</a> .
MaxPriceIsSupportedOnlyForAzureSpotVirtualMachines	Max price is supported only for Azure Spot Virtual Machines.	For more information, see <a href="#">Azure Spot Virtual Machines</a> .
MoveResourcesWithAzureSpotVMNotSupported	The Move resources request contains an Azure Spot Virtual Machine. Not supported. Check the error details for virtual machine Ids.	You cannot move Azure Spot Virtual Machines.
MoveResourcesWithAzureSpotVmssNotSupported	The Move resources request contains an Azure Spot virtual machine scale set. Not supported. Check the error details for virtual machine scale set Ids.	You cannot move Azure Spot virtual machine scale set instances.
AzureSpotVMNotSupportedInVmssWithVMOrchestrationMode	Azure Spot Virtual Machine is not supported in Virtual machine scale set with VM Orchestration mode.	Set the orchestration mode to virtual machine scale set in order to use Azure Spot Virtual Machine instances.
SpotRestorationIsNotSupportedForThisAPIVersion	Spot restoration feature is not supported for this API version.	<p>For an existing scaleset, perform a PATCH using API version 2021-07-01 or later.</p> <p>For new scale set deployments, add the following property to the Azure Resource Manager template using API version 2021-07-01 or later:</p> <pre>{   "properties":{     "spotRestorePolicy":{       "enabled":false,       "restoreTimeout":"PT48H"     }   } }</pre>
SpotRestorationIsSupportedOnlyForAzureSpotScaleSets	Spot restoration feature is supported only for Azure Spot virtual machine scale sets.	Spot restoration feature is only supported for Azure Spot virtual machine scale sets. To use this feature, deploy Azure Spot using virtual machine scale sets.

**Next steps** For more information, see [spot Virtual Machines](#).

# Explore Azure Hybrid Benefit for pay-as-you-go Linux virtual machines

9/21/2022 • 11 minutes to read • [Edit Online](#)

Azure Hybrid Benefit for pay-as-you-go virtual machines or virtual machine scale sets (Flexible orchestration mode only) is an optional licensing benefit. It significantly reduces the cost of running Red Hat Enterprise Linux (RHEL) and SUSE Linux Enterprise Server (SLES) virtual machines in the cloud.

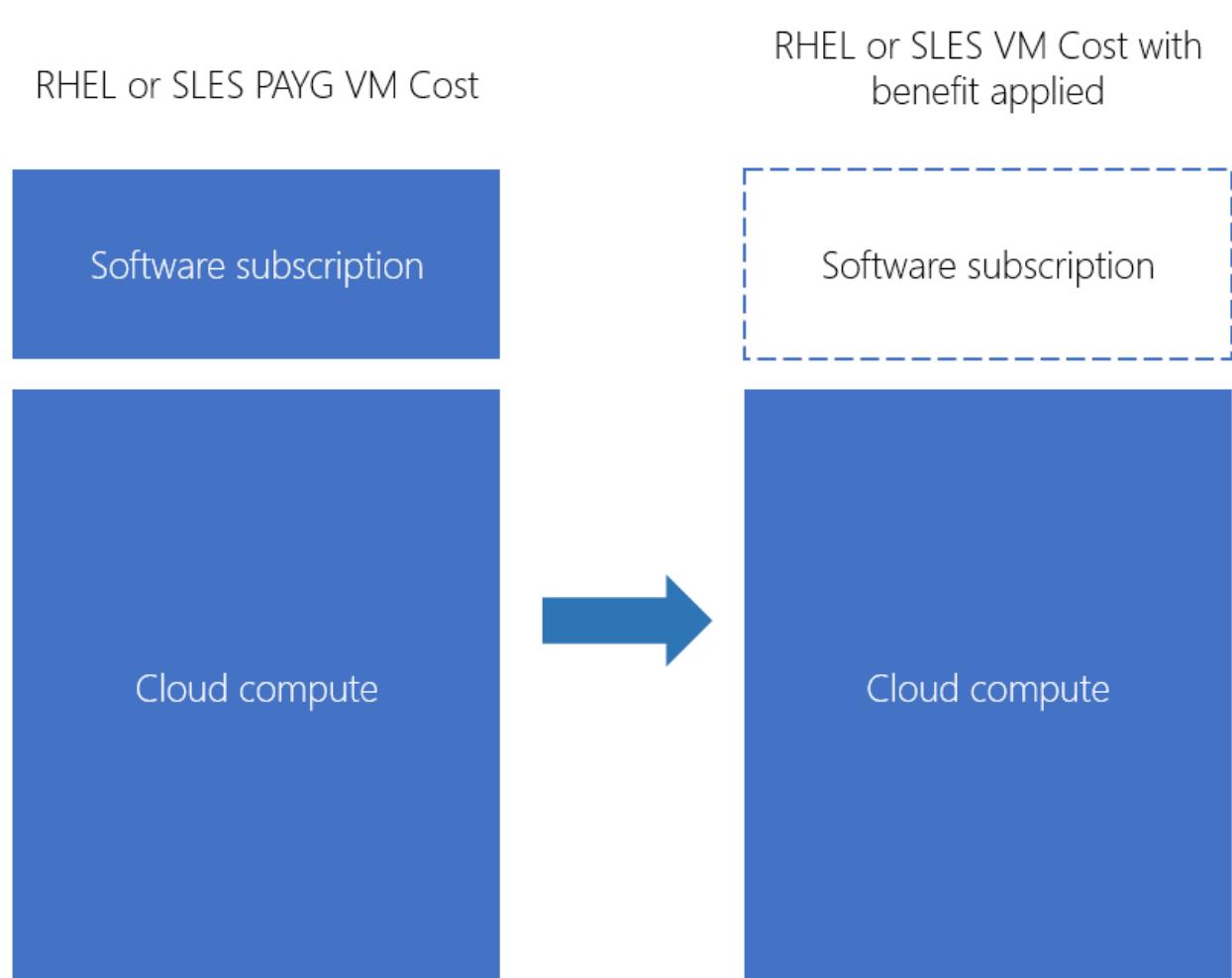
This article explores how to use Azure Hybrid Benefit to switch your virtual machines or virtual machine scale sets (Flexible orchestration mode only) to RHEL bring-your-own-subscription (BYOS) and SLES BYOS billing. With this benefit, your RHEL or SLES subscription covers your software fee. So you pay only infrastructure costs for your virtual machine.

## IMPORTANT

To do the reverse and switch from BYOS to pay-as-you-go billing, see [Explore Azure Hybrid Benefit for bring-your-own-subscription Linux virtual machines](#).

## How does Azure Hybrid Benefit work?

Virtual machines deployed from pay-as-you-go images on Azure incur both an infrastructure fee and a software fee. You can convert existing RHEL and SLES pay-as-you-go virtual machines to BYOS billing by using Azure Hybrid Benefit without having to redeploy.



After you apply Azure Hybrid Benefit to your RHEL or SLES virtual machine, you're no longer charged a software fee. Your virtual machine is charged a BYOS fee instead. You can use Azure Hybrid Benefit to switch back to pay-as-you-go billing at any time.

## Which Linux virtual machines qualify for Azure Hybrid Benefit?

Azure Hybrid Benefit for pay-as-you-go virtual machines is available for all RHEL and SLES pay-as-you-go images in Azure Marketplace.

Azure dedicated host instances and SQL hybrid benefits are not eligible for Azure Hybrid Benefit if you already use Azure Hybrid Benefit with Linux virtual machines.

## Get started

### Apply Azure Hybrid Benefit to Red Hat

Azure Hybrid Benefit for pay-as-you-go virtual machines for RHEL is available to Red Hat customers who meet the following criteria:

- Have active or unused RHEL subscriptions that are eligible for use in Azure
- Have correctly enabled one or more of their subscriptions for use in Azure with the [Red Hat Cloud Access](#) program

To start using Azure Hybrid Benefit for Red Hat:

1. Enable one or more of your eligible RHEL subscriptions for use in Azure by using the [Red Hat Cloud Access customer interface](#).

The Azure subscriptions that you provide during the Red Hat Cloud Access enablement process will then be permitted to use Azure Hybrid Benefit.

2. Apply Azure Hybrid Benefit to any RHEL pay-as-you-go virtual machines that you deploy in Azure Marketplace pay-as-you-go images. You can use the Azure portal or the Azure CLI to enable Azure Hybrid Benefit.
3. Follow the recommended [next steps](#) to configure update sources for your RHEL virtual machines and for RHEL subscription compliance guidelines.

### Apply Azure Hybrid Benefit to SUSE

Azure Hybrid Benefit for pay-as-you-go virtual machines for SUSE is available to customers who have:

- Unused SUSE subscriptions that are eligible to use in Azure.
- One or more active SUSE subscriptions to use on-premises that should be moved to Azure.
- Purchased subscriptions that they activated in the SUSE Customer Center to use in Azure.

#### IMPORTANT

Ensure that you select the correct subscription to use in Azure.

To start using Azure Hybrid Benefit for SUSE:

1. Register the subscription that you purchased from SUSE or a SUSE distributor with the [SUSE Customer Center](#).
2. Activate the subscription in the SUSE Customer Center.
3. Register your virtual machines that are receiving Azure Hybrid Benefit with the SUSE Customer Center to get the updates from the SUSE Customer Center.

# Enable Azure Hybrid Benefit in the Azure portal

In the Azure portal, you can enable Azure Hybrid Benefit on existing virtual machines or on new virtual machines at the time that you create them.

## Enable Azure Hybrid Benefit on an existing virtual machine in the Azure portal

To enable Azure Hybrid Benefit on an existing virtual machine:

1. Go to the [Azure portal](#).
2. Open the virtual machine page on which you want to apply the conversion.
3. Go to **Configuration > Licensing**. To enable the Azure Hybrid Benefit conversion, select **Yes**, and then select the confirmation checkbox.

The screenshot shows the Azure portal interface for managing a virtual machine. The left sidebar lists various settings like Overview, Activity log, and Configuration. The Configuration section is currently selected. In the main content area, under the 'Licensing' heading, there is a note about Red Hat Enterprise Linux subscriptions. Below this, there are two radio buttons for 'Would you like to use an existing Red Hat Enterprise Linux subscription?': 'Yes' (selected) and 'No'. A checkbox below the radio buttons is checked, stating 'I confirm I have an eligible Red Hat Enterprise Linux subscription to attach to this VM.' A red box highlights this checkbox. At the bottom of the configuration section, there is a note about proximity placement groups.

## Enable Azure Hybrid Benefit when you create a virtual machine in the Azure portal

To enable Azure Hybrid Benefit when you create a virtual machine, use the following procedure. (The SUSE workflow is the same as the RHEL example shown here.)

1. Go to the [Azure portal](#).
2. Go to [Create a virtual machine](#).



Home &gt; New &gt;

## Create a virtual machine

Username \* ⓘ  ✓

SSH public key source  ▾

Key pair name \*

**Inbound port rules**

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports \* ⓘ  None  Allow selected ports

Select inbound ports \*  ▾

**⚠ This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

### Licensing

If you have eligible Red Hat Enterprise Linux subscriptions that are enabled for Red Hat Cloud Access, you can use Azure Hybrid Benefit to attach your Red Hat subscriptions to this VM and save money on compute costs [Learn more ↗](#)

Would you like to use an existing Red Hat  Enterprise Linux subscription? \*

3. In the **Licensing** section, select the checkbox that asks if you want to use an existing RHEL subscription and the checkbox to confirm that your subscription is eligible.

### Licensing

If you have eligible Red Hat Enterprise Linux subscriptions that are enabled for Red Hat Cloud Access, you can use Azure Hybrid Benefit to attach your Red Hat subscriptions to this VM and save money on compute costs [Learn more ↗](#)

Would you like to use an existing Red Hat  Enterprise Linux subscription? \*

I confirm I have an eligible Red Hat Enterprise Linux subscription to attach to this VM. \*

4. Create a virtual machine by following the next set of instructions.

5. On the Configuration pane, confirm that the option is enabled.

Just-in-time VM access  
To improve security, enable just-in-time VM access.  
[Upgrade your Security Center subscription to enable a just-in-time access](#)

**Licensing**

If you have eligible Red Hat Enterprise Linux subscriptions that are enabled for Red Hat Cloud Access, you can use Azure Hybrid Benefit to attach your Red Hat subscriptions to this VM and save money on compute costs. [Learn more about Azure Hybrid Benefit](#)

Would you like to use an existing Red Hat Enterprise Linux subscription? \*

Yes  No

I confirm I have an eligible Red Hat Enterprise Linux subscription to attach to this VM. \*

Proximity placement group  
Proximity placement group ⓘ  
No proximity placement groups found

Proximity placement group can only be updated when the virtual machine is deallocated.

## Enable and disable Azure Hybrid Benefit by using the Azure CLI

You can use the `az vm update` command to update existing virtual machines. For RHEL virtual machines, run the command with a `--license-type` parameter of `RHEL_BYOS`. For SLES virtual machines, run the command with a `--license-type` parameter of `SLES_BYOS`.

### Enable Azure Hybrid Benefit by using the Azure CLI

```
# This will enable Azure Hybrid Benefit on a RHEL virtual machine
az vm update -g myResourceGroup -n myVmName --license-type RHEL_BYOS

# This will enable Azure Hybrid Benefit on a SLES virtual machine
az vm update -g myResourceGroup -n myVmName --license-type SLES_BYOS
```

### Disable Azure Hybrid Benefit by using the Azure CLI

To disable Azure Hybrid Benefit, use a `--license-type` value of `None`:

```
# This will disable Azure Hybrid Benefit on a virtual machine
az vm update -g myResourceGroup -n myVmName --license-type None
```

### Enable Azure Hybrid Benefit on a large number of virtual machines by using the Azure CLI

To enable Azure Hybrid Benefit on a large number of virtual machines, you can use the `--ids` parameter in the Azure CLI:

```
# This will enable Azure Hybrid Benefit on a RHEL virtual machine. In this example, ids.txt is an
# existing text file that contains a delimited list of resource IDs corresponding
# to the virtual machines using Azure Hybrid Benefit
az vm update -g myResourceGroup -n myVmName --license-type RHEL_BYOS --ids $(cat ids.txt)
```

The following examples show two methods of getting a list of resource IDs: one at the resource group level, and one at the subscription level.

```
# To get a list of all the resource IDs in a resource group:  
$(az vm list -g MyResourceGroup --query "[].id" -o tsv)  
  
# To get a list of all the resource IDs of virtual machines in a subscription:  
az vm list -o json | jq '.[] | {Virtual MachineName: .name, ResourceID: .id}'
```

## Apply Azure Hybrid Benefit to pay-as-you-go virtual machines at creation time

In addition to applying Azure Hybrid Benefit to existing pay-as-you-go virtual machines, you can invoke it at the time of virtual machine creation. Benefits of doing so are threefold:

- You can provision both pay-as-you-go and BYOS virtual machines by using the same image and process.
- It enables future licensing mode changes. These changes aren't available with a BYOS-only image or if you bring your own virtual machine.
- The virtual machine will be connected to Red Hat Update Infrastructure (RHUI) by default, to help keep it up to date and secure. You can change the updated mechanism after deployment at any time.

## Check the Azure Hybrid Benefit status of a virtual machine

You can view the Azure Hybrid Benefit status of a virtual machine by using the Azure CLI or by using Azure Instance Metadata Service.

### Check status by using the Azure CLI

You can use the `az vm get-instance-view` command to check the status. Look for a `licenseType` field in the response. If the `licenseType` field exists and the value is `RHEL_BYOS` or `SLES_BYOS`, your virtual machine has Azure Hybrid Benefit enabled.

```
az vm get-instance-view -g MyResourceGroup -n MyVm
```

### Check status by using Azure Instance Metadata Service

From within the virtual machine itself, you can query the attested metadata in Azure Instance Metadata Service to determine the virtual machine's `licenseType` value. A `licenseType` value of `RHEL_BYOS` or `SLES_BYOS` indicates that your virtual machine has Azure Hybrid Benefit enabled. [Learn more about attested metadata](#).

## Compliance

### Red Hat compliance

Customers who use Azure Hybrid Benefit for pay-as-you-go RHEL virtual machines agree to the standard [legal terms](#) and [privacy statement](#) associated with the Azure Marketplace RHEL offers.

Customers who use Azure Hybrid Benefit for pay-as-you-go RHEL virtual machines have three options for providing software updates and patches to those virtual machines:

- [Red Hat Update Infrastructure](#) (default option)
- Red Hat Satellite Server
- Red Hat Subscription Manager

Customers who choose the RHUI option can continue to use RHUI as the main update source for Azure Hybrid Benefit for pay-as-you-go RHEL virtual machines without attaching RHEL subscriptions to those virtual machines. Customers who choose the RHUI option are responsible for ensuring RHEL subscription compliance.

Customers who choose either Red Hat Satellite Server or Red Hat Subscription Manager should remove the

RHUI configuration and then attach a cloud-access-enabled RHEL subscription to Azure Hybrid Benefit for pay-as-you-go RHEL virtual machines.

For more information about Red Hat subscription compliance, software updates, and sources for Azure Hybrid Benefit for pay-as-you-go RHEL virtual machines, see the [Red Hat article about using RHEL subscriptions with Azure Hybrid Benefit](#).

## SUSE compliance

To use Azure Hybrid Benefit for pay-as-you-go SLES virtual machines, and to get information about moving from SLES pay-as-you-go to BYOS or moving from SLES BYOS to pay-as-you-go, see [SUSE Linux Enterprise and Azure Hybrid Benefit](#).

Customers who use Azure Hybrid Benefit for pay-as-you-go SLES virtual machines need to move the cloud update infrastructure to one of three options that provide software updates and patches to those virtual machines:

- [SUSE Customer Center](#)
- SUSE Manager
- SUSE Repository Mirroring Tool

## Apply Azure Hybrid Benefit for pay-as-you-go virtual machines on reserved instances

[Azure reservations](#) (Azure Reserved Virtual Machine Instances) help you save money by committing to one-year or three-year plans for multiple products. Azure Hybrid Benefit for pay-as-you-go virtual machines is available for reserved instances.

This means that if you've purchased compute costs at a discounted rate by using reserved instances, you can apply Azure Hybrid Benefit on the licensing costs for RHEL and SUSE on top of it. The steps to apply Azure Hybrid Benefit for a reserved instance remain exactly same as they are for a regular virtual machine.

The screenshot shows the Microsoft Azure portal interface. At the top, there's a blue header bar with the Microsoft Azure logo and a navigation menu. Below the header, the URL 'Home > Reservations >' is visible. The main title is 'Purchase reservations'. Underneath, there are two tabs: 'Products' (which is underlined in blue) and 'Review + buy'. A sub-section title 'Receive discounts on your Azure services by purchasing reservations. See FAQs' is followed by a 'Filter by name...' search bar. A list of service icons and names follows, with 'Virtual machine' highlighted with a red border. Other listed services include SQL Database, Azure Dedicated Host, Azure Managed Disks, SUSE Linux, App Services, Data Factory, Azure Red Hat OpenShift, Azure Data Explorer, Azure Database for PostgreSQL, Azure Blob Storage, and another 'Virtual machine' entry.

Service	Description
Virtual machine	Selected
Azure Blob Storage	
Azure Database for PostgreSQL	
Azure Data Explorer	
Azure Red Hat OpenShift	
Data Factory	
SQL Database	
Azure Dedicated Host	
Azure Managed Disks	
SUSE Linux	
App Services	

## NOTE

If you've already purchased reservations for RHEL or SUSE pay-as-you-go software on Azure Marketplace, please wait for the reservation tenure to finish before using Azure Hybrid Benefit for pay-as-you-go virtual machines.

## Frequently asked questions

*Q: Can I use a license type of `RHEL_BYOS` with a SLES image, or vice versa?*

A: No, you can't. Trying to enter a license type that incorrectly matches the distribution running on your virtual machine will not update any billing metadata. But if you accidentally enter the wrong license type, updating your virtual machine again to the correct license type will still enable Azure Hybrid Benefit.

*Q: I've registered with Red Hat Cloud Access but still can't enable Azure Hybrid Benefit on my RHEL virtual machines. What should I do?*

A: It might take some time for your Red Hat Cloud Access subscription registration to propagate from Red Hat to Azure. If you still see the error after one business day, contact Microsoft support.

*Q: I've deployed a virtual machine by using a RHEL BYOS "golden image." Can I convert the billing on this image from BYOS to pay-as-you-go?*

A: Yes, you can use Azure Hybrid Benefit for BYOS virtual machines to do this. [Learn more about this capability](#).

*Q: I've uploaded my own RHEL or SLES image from on-premises (via Azure Migrate, Azure Site Recovery, or otherwise) to Azure. Can I convert the billing on these images from BYOS to pay-as-you-go?*

A: Yes, you can use Azure Hybrid Benefit for BYOS virtual machines to do this. [Learn more about this capability](#).

*Q: I've uploaded my own RHEL or SLES image from on-premises (via Azure Migrate, Azure Site Recovery, or otherwise) to Azure. Do I need to do anything to benefit from Azure Hybrid Benefit?*

A: No, you don't. RHEL or SLES images that you upload are already considered BYOS, and you're charged only for Azure infrastructure costs. You're responsible for RHEL subscription costs, just as you are for your on-premises environments.

*Q: Can I use Azure Hybrid Benefit for pay-as-you-go virtual machines for Azure Marketplace RHEL and SLES SAP images?*

A: Yes. You can use the license type of `RHEL_BYOS` for RHEL virtual machines and `SLES_BYOS` for conversions of virtual machines deployed from Azure Marketplace RHEL and SLES SAP images.

*Q: Can I use Azure Hybrid Benefit for pay-as-you-go virtual machines on virtual machine scale sets for RHEL and SLES?*

A: Yes. Azure Hybrid Benefit on virtual machine scale sets for RHEL and SLES is available to all users. [Learn more about this benefit and how to use it](#).

*Q: Can I use Azure Hybrid Benefit for pay-as-you-go virtual machines on reserved instances for RHEL and SLES?*

A: Yes. Azure Hybrid Benefit for pay-as-you-go virtual machines on reserved instances for RHEL and SLES is available to all users.

*Q: Can I use Azure Hybrid Benefit for pay-as-you-go virtual machines on a virtual machine deployed for SQL Server on RHEL images?*

A: No, you can't. There's no plan for supporting these virtual machines.

*Q: Can I use Azure Hybrid Benefit on my RHEL for Virtual Datacenters subscription?*

A: No. RHEL for Virtual Datacenters isn't supported on Azure at all, including Azure Hybrid Benefit.

## Common problems

This section lists common problems that you might encounter and steps for mitigation.

ERROR	MITIGATION
"The action could not be completed because our records show that you have not successfully enabled Red Hat Cloud Access on your Azure subscription."	To use Azure Hybrid Benefit with RHEL virtual machines, you must first <a href="#">register your Azure subscriptions with Red Hat Cloud Access</a> .

## Next steps

- [Learn how to create and update virtual machines and add license types \(RHEL\\_BYOS, SLES\\_BYOS\) for Azure Hybrid Benefit by using the Azure CLI](#)
- [Learn about Azure Hybrid Benefit on virtual machine scale sets for RHEL and SLES and how to use it](#)

# Explore Azure Hybrid Benefit for bring-your-own-subscription Linux virtual machines

9/21/2022 • 10 minutes to read • [Edit Online](#)

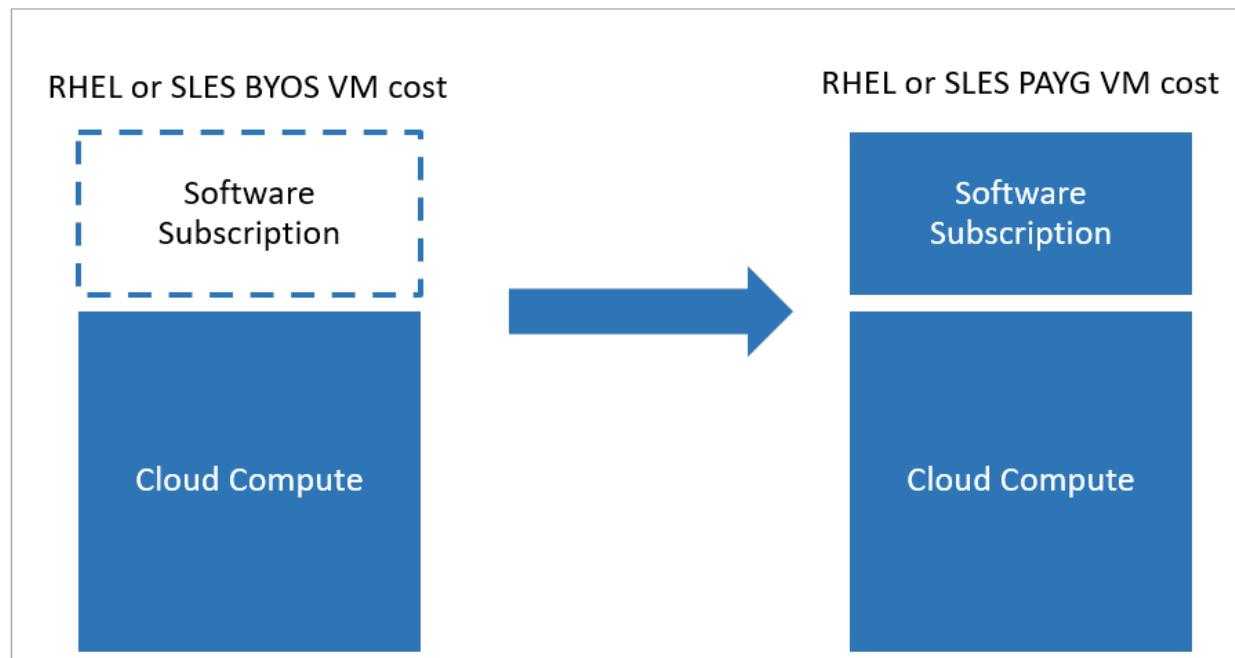
Azure Hybrid Benefit provides software updates and integrated support directly from Azure for Red Hat Enterprise Linux (RHEL) and SUSE Linux Enterprise Server (SLES) virtual machines. Azure Hybrid Benefit for bring-your-own-subscription (BYOS) virtual machines is a licensing benefit that lets you switch RHEL and SLES BYOS virtual machines generated from custom on-premises images or from Azure Marketplace to pay-as-you-go billing.

## IMPORTANT

To do the reverse and switch from a RHEL pay-as-you-go virtual machine or SLES pay-as-you-go virtual machine to a BYOS virtual machine, see [Explore Azure Hybrid Benefit for pay-as-you-go Linux virtual machines](#).

## How does Azure Hybrid Benefit work?

Azure Hybrid Benefit converts BYOS billing to pay-as-you-go, so that you pay only pay-as-you-go software fees. You don't have to restart a machine for Azure Hybrid Benefit to be applied.



## Which Linux virtual machines qualify for Azure Hybrid Benefit?

Azure Hybrid Benefit for BYOS virtual machines is available to all RHEL and SLES virtual machines that come from a custom image. It's also available to all RHEL and SLES BYOS virtual machines that come from an Azure Marketplace image.

Azure Dedicated Host instances and SQL hybrid benefits aren't eligible for Azure Hybrid Benefit if you're already using Azure Hybrid Benefit with Linux virtual machines. Virtual machine scale sets are reserved instances, so they also can't use Azure Hybrid Benefit for BYOS virtual machines.

# Get started

## Azure Hybrid Benefit for Red Hat customers

To start using Azure Hybrid Benefit for Red Hat:

1. Install the **AHBForRHEL** extension on the virtual machine on which you want to apply the Azure Hybrid Benefit BYOS benefit. You can do this installation via the Azure CLI or PowerShell.
2. Depending on the software updates that you want, change the license type to a relevant value. Here are the available license type values and the software updates associated with them:

LICENSE TYPE	SOFTWARE UPDATES	ALLOWED VIRTUAL MACHINES
RHEL_BASE	Installs Red Hat regular/base repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines
RHEL_EUS	Installs Red Hat Extended Update Support (EUS) repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines
RHEL_SAPAPPS	Installs RHEL for SAP Business Apps repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines
RHEL_SAPHA	Installs RHEL for SAP with High Availability (HA) repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines
RHEL_BASESAPAPPS	Installs RHEL regular/base SAP Business Apps repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines
RHEL_BASESAPHA	Installs regular/base RHEL for SAP with HA repositories on your virtual machine.	RHEL BYOS virtual machines, RHEL custom image virtual machines

3. Wait one hour for the extension to read the license type value and install the repositories.

### NOTE

If the extension isn't running by itself, you can run it on demand.

4. You should now be connected to Azure Red Hat Update. The relevant repositories will be installed on your machine.
5. If you want to switch back to the bring-your-own-subscription model, just change the license type to **None** and run the extension. This action will remove all Red Hat Update Infrastructure (RHUI) repositories from your virtual machine and stop the billing.

### NOTE

In the unlikely event that the extension can't install repositories or there are any other issues, switch the license type back to empty and reach out to Microsoft support. This ensures that you don't get billed for software updates.

## Azure Hybrid Benefit for SUSE customers

To start using Azure Hybrid Benefit for SLES virtual machines:

1. Install the `AHBForSLES` extension on the virtual machine that will use it.
2. Change the license type to the value that reflects the software updates you want. Here are the available license type values and the software updates associated with them:

LICENSE TYPE	SOFTWARE UPDATES	ALLOWED VIRTUAL MACHINES
SLES	Installs SLES Standard repositories on your virtual machine.	SLES BYOS virtual machines, SLES custom image virtual machines
SLES_SAP	Installs SLES SAP repositories on your virtual machine.	SLES SAP BYOS virtual machines, SLES custom image virtual machines
SLES_HPC	Installs SLES High Performance Computing repositories on your virtual machine.	SLES HPC BYOS virtual machines, SLES custom image virtual machines

3. Wait five minutes for the extension to read the license type value and install the repositories.

**NOTE**

If the extension isn't running by itself, you can run it on demand.

4. You should now be connected to the SUSE public cloud update infrastructure on Azure. The relevant repositories will be installed on your machine.
5. If you want to switch back to the bring-your-own-subscription model, just change the license type to `None` and run the extension. This action will remove all repositories from your virtual machine and stop the billing.

## Enable Azure Hybrid Benefit for RHEL

After you successfully install the `AHBForRHEL` extension, you can use the `az vm update` command to update the existing license type on your running virtual machines. For SLES virtual machines, run the command and set the `--license-type` parameter to one of the following license types: `RHEL_BASE`, `RHEL_EUS`, `RHEL_SAPHA`, `RHEL_SAPAPP`, `RHEL_BASESAPAPP`, or `RHEL_BASESAPHA`.

### Enable Azure Hybrid Benefit for RHEL by using the Azure CLI

1. Install the Azure Hybrid Benefit extension on a running virtual machine. You can use the Azure portal or use the following command via the Azure CLI:

```
az vm extension set -n AHBForRHEL --publisher Microsoft.Azure.AzureHybridBenefit --vm-name myVMName --resource-group myResourceGroup
```

2. After the extension is installed successfully, change the license type based on what you need:

```
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL base/regular repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_BASE  
  
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL EUS repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_EUS  
  
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL SAP APPS repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_SAPAPPS  
  
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL SAP HA repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_SAPHA  
  
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL BASE SAP APPS repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_BASESAPAPPS  
  
# This will enable Azure Hybrid Benefit to fetch software updates for RHEL BASE SAP HA repositories  
az vm update -g myResourceGroup -n myVmName --license-type RHEL_BASESAPHA
```

3. Wait five minutes for the extension to read the license type value and install the repositories.
4. You should now be connected to Red Hat Update Infrastructure. The relevant repositories will be installed on your machine. You can validate the installation by running the following command on your virtual machine:

```
yum repolist
```

5. If the extension isn't running by itself, you can try the following command on the virtual machine:

```
systemctl start azure-hybrid-benefit.service
```

6. You can use the following command in your RHEL virtual machine to get the current status of the service:

```
ahb-service -status
```

## Enable and disable Azure Hybrid Benefit for SLES

After you successfully install the `AHBForSLES` extension, you can use the `az vm update` command to update the existing license type on your running virtual machines. For SLES virtual machines, run the command and set the `--license-type` parameter to one of the following license types: `SLES_STANDARD`, `SLES_SAP`, or `SLES_HPC`.

### Enable Azure Hybrid Benefit for SLES by using the Azure CLI

1. Install the Azure Hybrid Benefit extension on a running virtual machine. You can use the Azure portal or use the following command via the Azure CLI:

```
az vm extension set -n AHBForSLES --publisher SUSE.AzureHybridBenefit --vm-name myVMName --resource-group myResourceGroup
```

2. After the extension is installed successfully, change the license type based on what you need:

```
# This will enable Azure Hybrid Benefit to fetch software updates for SLES Standard repositories  
az vm update -g myResourceGroup -n myVmName --license-type SLES  
  
# This will enable Azure Hybrid Benefit to fetch software updates for SLES SAP repositories  
az vm update -g myResourceGroup -n myVmName --license-type SLES_SAP  
  
# This will enable Azure Hybrid Benefit to fetch software updates for SLES HPC repositories  
az vm update -g myResourceGroup -n myVmName --license-type SLES_HPC
```

3. Wait five minutes for the extension to read the license type value and install the repositories.
4. You should now be connected to the SUSE public cloud update infrastructure on Azure. The relevant repositories will be installed on your machine. You can verify this change by running the following command to list SUSE repositories on your machine:

```
zypper repos
```

### Disable Azure Hybrid Benefit by using the Azure CLI

1. Ensure that the Azure Hybrid Benefit extension is installed on your virtual machine.
2. To disable Azure Hybrid Benefit, use the following command:

```
# This will disable Azure Hybrid Benefit on a virtual machine  
az vm update -g myResourceGroup -n myVmName --license-type None
```

## Check the Azure Hybrid Benefit status of a virtual machine

1. Ensure that the Azure Hybrid Benefit extension is installed.
2. In the Azure CLI or Azure Instance Metadata Service, run the following command:

```
az vm get-instance-view -g MyResourceGroup -n MyVm
```

3. Look for a `licenseType` field in the response. If the `licenseType` field exists and the value is one of the following, your virtual machine has Azure Hybrid Benefit enabled:

`RHEL_BASE` , `RHEL_EUS` , `RHEL_BASESAPAPPS` , `RHEL_SAPHA` , `RHEL_BASESAPAPPS` , `RHEL_BASESAPHA` , `SLES` ,  
`SLES_SAP` , `SLES_HPC`

## Compliance

### Red Hat compliance

Customers who use Azure Hybrid Benefit for BYOS virtual machines for RHEL agree to the standard [legal terms](#) and [privacy statement](#) associated with the Azure Marketplace RHEL offerings.

### SUSE compliance

If you use Azure Hybrid Benefit for BYOS virtual machines for SLES and want more information about moving from SLES pay-as-you-go to BYOS, or moving from SLES BYOS to pay-as-you-go, see [Azure Hybrid Benefit Support](#) on the SUSE website.

## Frequently asked questions

*Q: What is the licensing cost I pay with Azure Hybrid Benefit for BYOS virtual machines?*

A: When you start using Azure Hybrid Benefit for BYOS virtual machines, you'll essentially convert the bring-your-own-subscription billing model to a pay-as-you-go billing model. What you pay will be similar to a software subscription cost for pay-as-you-go virtual machines.

The following table maps the pay-as-you-go options on Azure and links to pricing information to help you understand the cost associated with Azure Hybrid Benefit for BYOS virtual machines. When you go to the pricing pages, keep the Azure Hybrid Benefit for pay-as-you-go filter off.

LICENSE TYPE	RELEVANT PAY-AS-YOU-GO VIRTUAL MACHINE IMAGE AND PRICING LINK
RHEL_BASE	<a href="#">Red Hat Enterprise Linux</a>
RHEL_SAPAPPS	<a href="#">RHEL for SAP Business Applications</a>
RHEL_SAPHA	<a href="#">RHEL for SAP with HA</a>
RHEL_BASESAPAPPS	<a href="#">RHEL for SAP Business Applications</a>
RHEL_BASESAPHA	<a href="#">RHEL for SAP with HA</a>
RHEL_EUS	<a href="#">Red Hat Enterprise Linux</a>
SLES	<a href="#">SLES</a>
SLES_SAP	<a href="#">SLES for SAP</a>
SLES_HPC	<a href="#">SLE HPC</a>

*Q: Can I use a license type designated for RHEL (such as `RHEL_BASE`) with a SLES image, or vice versa?*

A: No, you can't. Trying to enter a license type that incorrectly matches the distribution running on your virtual machine will fail, and you might end up getting billed incorrectly.

If you accidentally enter the wrong license type, remove the billing by changing the license type to empty. Then update your virtual machine to the correct license type to enable Azure Hybrid Benefit.

*Q: What are the supported versions for RHEL with Azure Hybrid Benefit for BYOS virtual machines?*

A: Azure Hybrid Benefit for BYOS virtual machines supports RHEL versions later than 7.4.

*Q: I've uploaded my own RHEL or SLES image from on-premises (via Azure Migrate, Azure Site Recovery, or otherwise) to Azure. Can I convert the billing on these images from BYOS to pay-as-you-go?*

A: Yes, this capability supports images uploaded from on-premises to Azure. Follow the steps in the [Get started](#) section earlier in this article.

*Q: Can I use Azure Hybrid Benefit for BYOS virtual machines on RHEL and SLES pay-as-you-go Azure Marketplace virtual machines?*

A: No, because these virtual machines are already pay-as-you-go. However, with Azure Hybrid Benefit, you can use the license type of `RHEL_BYOS` for RHEL virtual machines and `SLES_BYOS` for conversions of RHEL and SLES pay-as-you-go Azure Marketplace virtual machines. For more information, see [Explore Azure Hybrid Benefit for pay-as-you-go Linux virtual machines](#).

*Q: Can I use Azure Hybrid Benefit for BYOS virtual machines on virtual machine scale sets for RHEL and SLES?*

A: No. Hybrid Benefit for BYOS virtual machines isn't currently available for virtual machine scale sets.

*Q: Can I use Azure Hybrid Benefit for BYOS virtual machines on a virtual machine deployed for SQL Server on RHEL images?*

A: No, you can't. There's no plan for supporting these virtual machines.

*Q: Can I use Azure Hybrid Benefit for BYOS virtual machines on my RHEL for Virtual Datacenters subscription?*

A: No. RHEL for Virtual Datacenters isn't supported on Azure at all, including Azure Hybrid Benefit.

## Next steps

- [Learn how to convert RHEL and SLES pay-as-you-go virtual machines to BYOS by using Azure Hybrid Benefit](#)
- [Learn how to create and update virtual machines and add license types \(RHEL\\_BYOS, SLES\\_BYOS\) for Azure Hybrid Benefit by using the Azure CLI](#)

# Azure Hybrid Benefit for Windows Server

9/21/2022 • 5 minutes to read • [Edit Online](#)

For customers with Software Assurance, Azure Hybrid Benefit for Windows Server allows you to use your on-premises Windows Server licenses and run Windows virtual machines on Azure at a reduced cost. You can use Azure Hybrid Benefit for Windows Server to deploy new virtual machines with Windows OS. This article goes over the steps on how to deploy new VMs with Azure Hybrid Benefit for Windows Server and how you can update existing running VMs. For more information about Azure Hybrid Benefit for Windows Server licensing and cost savings, see the [Azure Hybrid Benefit for Windows Server licensing page](#).

Each 2-processor license or each set of 16-core licenses is entitled to two instances of up to 8 cores, or one instance of up to 16 cores. The Azure Hybrid Benefit for Standard Edition licenses can only be used once either on-premises or in Azure. Datacenter Edition benefits allow for simultaneous usage both on-premises and in Azure.

Using Azure Hybrid Benefit for Windows Server with any VMs running Windows Server OS are now supported in all regions, including VMs with additional software such as SQL Server or third-party marketplace software.

## Classic VMs

For classic VMs, only deploying new VM from on premises custom images is supported. To take advantage of the capabilities supported in this article, you must first migrate classic VMs to Resource Manager model.

### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

## Ways to use Azure Hybrid Benefit for Windows Server

There are few ways to use Windows virtual machines with the Azure Hybrid Benefit:

1. You can deploy VMs from one of the provided Windows Server images on the Azure Marketplace
2. You can upload a custom VM and deploy using a Resource Manager template or Azure PowerShell
3. You can toggle and convert existing VM between running with Azure Hybrid Benefit or pay on-demand cost for Windows Server
4. You can also apply Azure Hybrid Benefit for Windows Server on virtual machine scale set as well!

## Create a VM with Azure Hybrid Benefit for Windows Server

All Windows Server OS based images are supported for Azure Hybrid Benefit for Windows Server. You can use Azure platform support images or upload your own custom Windows Server images.

### Portal

To create a VM with Azure Hybrid Benefit for Windows Server, scroll to the bottom of the **Basics** tab during the creation process and under **Licensing** check the box to use an existing Windows Server license.

## PowerShell

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroup" ` 
    -Name "myVM" ` 
    -Location "East US" ` 
    -ImageName "Win2016Datacenter" ` 
    -LicenseType "Windows_Server"
```

## CLI

```
az vm create \
    --resource-group myResourceGroup \
    --name myVM \
    --location eastus \
    --license-type Windows_Server
```

## Template

Within your Resource Manager templates, an additional parameter `licenseType` must be specified. You can read more about [authoring Azure Resource Manager templates](#).

```
"properties": { 
    "licenseType": "Windows_Server", 
    "hardwareProfile": { 
        "vmSize": "[variables('vmSize')]" 
    } 
}
```

# Convert an existing VM using Azure Hybrid Benefit for Windows Server

If you have an existing VM that you would like to convert to take advantage of Azure Hybrid Benefit for Windows Server, you can update your VM's license type by following the instructions below.

### NOTE

Changing the license type on the VM does not cause the system to reboot or cause a service interruption. It is simply an update to a metadata flag.

## Portal

From portal VM blade, you can update the VM to use Azure Hybrid Benefit by selecting "Configuration" option and toggle the "Azure hybrid benefit" option

## PowerShell

- Convert existing Windows Server VMs to Azure Hybrid Benefit for Windows Server

```
$vm = Get-AzVM -ResourceGroup "rg-name" -Name "vm-name"
$vm.LicenseType = "Windows_Server"
Update-AzVM -ResourceGroupName rg-name -VM $vm
```

- Convert Windows Server VMs with benefit back to pay-as-you-go

```
$vm = Get-AzVM -ResourceGroup "rg-name" -Name "vm-name"  
$vm.LicenseType = "None"  
Update-AzVM -ResourceGroupName rg-name -VM $vm
```

## CLI

- Convert existing Windows Server VMs to Azure Hybrid Benefit for Windows Server

```
az vm update --resource-group myResourceGroup --name myVM --set licenseType=Windows_Server
```

## How to verify your VM is utilizing the licensing benefit

Once you've deployed your VM through either PowerShell, Resource Manager template or portal, you can verify the setting in the following methods.

### Portal

From portal VM blade, you can view the toggle for Azure Hybrid Benefit for Windows Server by selecting "Configuration" tab.

### PowerShell

The following example shows the license type for a single VM

```
Get-AzVM -ResourceGroup "myResourceGroup" -Name "myVM"
```

Output:

```
Type : Microsoft.Compute/virtualMachines  
Location : westus  
LicenseType : Windows_Server
```

This output contrasts with the following VM deployed without Azure Hybrid Benefit for Windows Server licensing:

```
Type : Microsoft.Compute/virtualMachines  
Location : westus  
LicenseType :
```

## CLI

```
az vm get-instance-view -g MyResourceGroup -n MyVM --query "[?licenseType=='Windows_Server']" -o table
```

### NOTE

Changing the license type on the VM does not cause the system to reboot or cause a service interruption. It is a metadata licensing flag only.

## List all VMs and virtual machine scale sets with Azure Hybrid Benefit for Windows Server in a subscription

To see and count all virtual machines and virtual machine scale sets deployed with Azure Hybrid Benefit for Windows Server, you can run the following command from your subscription:

## Portal

From the Virtual Machine or Virtual machine scale sets resource blade, you can view a list of all your VM(s) and licensing type by configuring the table column to include "OS licensing benefit". The VM setting can either be in **Azure Hybrid Benefit for Windows**, **Not enabled**, or **Windows client with multi-tenant hosting** state.

## PowerShell

For virtual machines:

```
Get-AzVM | ?{$_.LicenseType -like "Windows_Server"} | select ResourceGroupName, Name, LicenseType
```

For virtual machine scale sets:

```
Get-AzVmss | Select * -ExpandProperty VirtualMachineProfile | ? LicenseType -eq 'Windows_Server' | select ResourceGroupName, Name, LicenseType
```

## CLI

For virtual machines:

```
az vm list --query "[?licenseType=='Windows_Server']" -o table
```

For virtual machine scale sets:

```
az vmss list --query "[?virtualMachineProfile.licenseType=='Windows_Server']" -o table
```

## Deploy a Virtual Machine Scale Set with Azure Hybrid Benefit for Windows Server

Within your virtual machine scale set Resource Manager templates, an additional parameter `licenseType` must be specified within your `VirtualMachineProfile` property. You can do this during create or update for your scale set through ARM template, PowerShell, Azure CLI or REST.

The following example uses ARM template with a Windows Server 2016 Datacenter image:

```
"virtualMachineProfile": {
    "storageProfile": {
        "osDisk": {
            "createOption": "FromImage"
        },
        "imageReference": {
            "publisher": "MicrosoftWindowsServer",
            "offer": "WindowsServer",
            "sku": "2016-Datacenter",
            "version": "latest"
        }
    },
    "licenseType": "Windows_Server",
    "osProfile": {
        "computerNamePrefix": "[parameters('vmssName')]",
        "adminUsername": "[parameters('adminUsername')]",
        "adminPassword": "[parameters('adminPassword')]"
    }
}
```

You can also learn more about how to [Modify a virtual machine scale set](#) for more ways to update your scale set.

## Next steps

- Read more about [How to save money with the Azure Hybrid Benefit](#)
- Read more about [Frequently asked questions for Azure Hybrid Benefit](#)
- Learn more about [Azure Hybrid Benefit for Windows Server licensing detailed guidance](#)
- Learn more about [Azure Hybrid Benefit for Windows Server and Azure Site Recovery make migrating applications to Azure even more cost-effective](#)
- Learn more about [Windows 10 on Azure with Multitenant Hosting Right](#)
- Learn more about [Using Resource Manager templates](#)

# What are Azure Reservations?

9/21/2022 • 6 minutes to read • [Edit Online](#)

Azure Reservations help you save money by committing to one-year or three-year plans for multiple products. Committing allows you to get a discount on the resources you use. Reservations can significantly reduce your resource costs by up to 72% from pay-as-you-go prices. Reservations provide a billing discount and don't affect the runtime state of your resources. After you purchase a reservation, the discount automatically applies to matching resources.

You can pay for a reservation up front or monthly. The total cost of up-front and monthly reservations is the same and you don't pay any extra fees when you choose to pay monthly. Monthly payment is available for Azure reservations, not third-party products.

You can buy a reservation in the [Azure portal](#).

## Why buy a reservation?

If you have consistent resource usage that supports reservations, buying a reservation gives you the option to reduce your costs. For example, when you continuously run instances of a service without a reservation, you're charged at pay-as-you-go rates. When you buy a reservation, you immediately get the reservation discount. The resources are no longer charged at the pay-as-you-go rates.

## How reservation discount is applied

After purchase, the reservation discount automatically applies to the resource usage that matches the attributes you select when you buy the reservation. Attributes include the SKU, regions (where applicable), and scope. Reservation scope selects where the reservation savings apply.

For more information about how discount is applied, see [Reserved instance discount application](#).

For more information about how reservation scope works, see [Scope reservations](#).

## Determine what to purchase

All reservations, except Azure Databricks, are applied on an hourly basis. Consider reservation purchases based on your consistent base usage. You can determine which reservation to purchase by analyzing your usage data or by using reservation recommendations. Recommendations are available in:

- Azure Advisor (VMs only)
- Reservation purchase experience in the Azure portal
- Cost Management Power BI app
- APIs

For more information, see [Determine what reservation to purchase](#)

## Buying a reservation

You can purchase reservations from the Azure portal, APIs, PowerShell, and CLI.

Go to the [Azure portal](#) to make a purchase.

For more information, see [Buy a reservation](#).

## How is a reservation billed?

The reservation is charged to the payment method tied to the subscription. The reservation cost is deducted from your Azure Prepayment (previously called monetary commitment) balance, if available. When your Azure Prepayment balance doesn't cover the cost of the reservation, you're billed the overage. If you have a subscription from an individual plan with pay-as-you-go rates, the credit card you have on your account is billed immediately for up-front purchases. Monthly payments appear on your invoice and your credit card is charged monthly. When you're billed by invoice, you see the charges on your next invoice.

## Who can manage a reservation by default

By default, the following users can view and manage reservations:

- The person who buys a reservation and the account administrator of the billing subscription used to buy the reservation are added to the reservation order.
- Enterprise Agreement and Microsoft Customer Agreement billing administrators.

To allow other people to manage reservations, see [Manage Reservations for Azure resources](#).

## Get reservation details and utilization after purchase

If you have permission to view to the reservation, you can see it and its use in the Azure portal. You can get the data using APIs, as well.

For more information on how to see reservations in Azure portal, see [View reservations in the Azure portal](#)

## Manage reservations after purchase

After you buy an Azure reservation, you can update the scope to apply reservation to a different subscription, change who can manage the reservation, split a reservation into smaller parts, or change instance size flexibility.

For more information, see [Manage Reservations for Azure resources](#)

## Flexibility with Azure reservations

Azure Reservations provide flexibility to help meet your evolving needs. You can exchange a reservation for another reservation of the same type. You can also refund a reservation, up to \$50,000 USD in a 12 month rolling window, if you no longer need it. The maximum limit of the refund applies to all reservations in the scope of your agreement with Microsoft.

For more information, see [Self-service exchanges and refunds for Azure Reservations](#)

## Charges covered by reservation

- **Reserved Virtual Machine Instance** - A reservation only covers the virtual machine and cloud services compute costs. It doesn't cover additional software, Windows, networking, or storage charges.
- **Azure Storage reserved capacity** - A reservation covers storage capacity for standard storage accounts for Blob storage or Azure Data Lake Gen2 storage. The reservation doesn't cover bandwidth or transaction rates.
- **Azure Cosmos DB reserved capacity** - A reservation covers throughput provisioned for your resources. It doesn't cover the storage and networking charges.
- **Azure Data Factory data flows** - A reservation covers integration runtime cost for the compute type and number of cores that you buy.
- **SQL Database reserved vCore** - Covers both SQL Managed Instance and SQL Database Elastic Pool/single database. Only the compute costs are included with a reservation. The SQL license is billed

separately.

- **Azure Synapse Analytics** - A reservation covers cDWU usage. It doesn't cover storage or networking charges associated with the Azure Synapse Analytics usage.
- **Azure Databricks** - A reservation covers only the DBU usage. Other charges, such as compute, storage, and networking, are applied separately.
- **App Service stamp fee** - A reservation covers stamp usage. It doesn't apply to workers, so any other resources associated with the stamp are charged separately.
- **Azure Database for MySQL** - Only the compute costs are included with a reservation. A reservation doesn't cover software, networking, or storage charges associated with the MySQL Database server.
- **Azure Database for PostgreSQL** - Only the compute costs are included with a reservation. A reservation doesn't cover software, networking, or storage charges associated with the PostgreSQL Database servers.
- **Azure Database for MariaDB** - Only the compute costs are included with a reservation. A reservation doesn't cover software, networking, or storage charges associated with the MariaDB Database server.
- **Azure Data Explorer** - A reservation covers the markup charges. A reservation doesn't apply to compute, networking, or storage charges associated with the clusters.
- **Azure Cache for Redis** - Only the compute costs are included with a reservation. A reservation doesn't cover networking or storage charges associated with the Redis cache instances.
- **Azure Dedicated Host** - Only the compute costs are included with the Dedicated host.
- **Azure Disk Storage reservations** - A reservation only covers premium SSDs of P30 size or greater. It doesn't cover any other disk types or sizes smaller than P30.
- **Azure Backup Storage reserved capacity** - A capacity reservation lowers storage costs of backup data in a Recovery Services Vault.

Software plans:

- **SUSE Linux** - A reservation covers the software plan costs. The discounts apply only to SUSE meters and not to the virtual machine usage.
- **Red Hat Plans** - A reservation covers the software plan costs. The discounts apply only to RedHat meters and not to the virtual machine usage.
- **Azure VMware Solution by CloudSimple** - A reservation covers the VMware CloudSimple Nodes. Additional software costs still apply.
- **Azure Red Hat OpenShift** - A reservation applies to the OpenShift costs, not to Azure infrastructure costs.

For Windows virtual machines and SQL Database, the reservation discount doesn't apply to the software costs. You can cover the licensing costs with [Azure Hybrid Benefit](#).

## Need help? Contact us.

If you have questions or need help, [create a support request](#).

## Next steps

- Learn more about Azure Reservations with the following articles:
  - [Manage Azure Reservations](#)
  - [Understand reservation usage for your subscription with pay-as-you-go rates](#)
  - [Understand reservation usage for your Enterprise enrollment](#)
  - [Windows software costs not included with reservations](#)
  - [Azure Reservations in Partner Center Cloud Solution Provider \(CSP\) program](#)
- Learn more about reservations for service plans:
  - [Virtual Machines with Azure Reserved VM Instances](#)

- [Azure Cosmos DB resources with Azure Cosmos DB reserved capacity](#)
- [SQL Database compute resources with Azure SQL Database reserved capacity](#)
- [Azure Cache for Redis resources with Azure Cache for Redis reserved capacity](#) Learn more about reservations for software plans:
  - [Red Hat software plans from Azure Reservations](#)
  - [SUSE software plans from Azure Reservations](#)

# Save costs with Azure Reserved VM Instances

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

When you commit to an Azure reserved VM instance you can save money. The reservation discount is applied automatically to the number of running virtual machines that match the reservation scope and attributes. You don't need to assign a reservation to a virtual machine to get the discounts. A reserved instance purchase covers only the compute part of your VM usage. For Windows VMs, the usage meter is split into two separate meters. There's a compute meter, which is same as the Linux meter, and a Windows IP meter. The charges that you see when you make the purchase are only for the compute costs. Charges don't include Windows software costs. For more information about software costs, see [Software costs not included with Azure Reserved VM Instances](#).

## Determine the right VM size before you buy

Before you buy a reservation, you should determine the size of the VM that you need. The following sections will help you determine the right VM size.

### Use reservation recommendations

You can use reservation recommendations to help determine the reservations you should purchase.

- Purchase recommendations and recommended quantities are shown when you purchase a VM reserved instance in the Azure portal.
- Azure Advisor provides purchase recommendations for individual subscriptions.
- You can use the APIs to get purchase recommendations for both shared scope and single subscription scope. For more information, see [Reserved instance purchase recommendation APIs for enterprise customers](#).
- For Enterprise Agreement (EA) and Microsoft Customer Agreement (MCA) customers, purchase recommendations for shared and single subscription scopes are available with the [Azure Consumption Insights Power BI content pack](#).

### Services that get VM reservation discounts

Your VM reservations can apply to VM usage emitted from multiple services - not just for your VM deployments. Resources that get reservation discounts change depending on the instance size flexibility setting.

#### Instance size flexibility setting

The instance size flexibility setting determines which services get the reserved instance discounts.

Whether the setting is on or off, reservation discounts automatically apply to any matching VM usage when the *ConsumedService* is `Microsoft.Compute`. So, check your usage data for the *ConsumedService* value. Some examples include:

- Virtual machines
- Virtual machine scale sets
- Container service
- Azure Batch deployments (in user subscriptions mode)
- Azure Kubernetes Service (AKS)
- Service Fabric

When the setting is on, reservation discounts automatically apply to matching VM usage when the *ConsumedService* is any of the following items:

- Microsoft.Compute
- Microsoft.ClassicCompute
- Microsoft.Batch
- Microsoft.MachineLearningServices
- Microsoft.Kusto

Check the *ConsumedService* value in your usage data to determine if the usage is eligible for reservation discounts.

For more information about instance size flexibility, see [Virtual machine size flexibility with Reserved VM Instances](#).

### Analyze your usage information

Analyze your usage information to help determine which reservations you should purchase. Usage data is available in the usage file and APIs. Use them together to determine which reservation to purchase. Check for VM instances that have high usage on daily basis to determine the quantity of reservations to purchase. Avoid the `Meter` subcategory and `Product` fields in usage data. They don't distinguish between VM sizes that use premium storage. If you use these fields to determine the VM size for reservation purchase, you may buy the wrong size. Then you won't get the reservation discount you expect. Instead, refer to the `AdditionalInfo` field in your usage file or usage API to determine the correct VM size.

Your usage file shows your charges by billing period and daily usage. For information about downloading your usage file, see [View and download your Azure usage and charges](#). Then, by using the usage file information, you can [determine what reservation to purchase](#).

### Purchase restriction considerations

Reserved VM Instances are available for most VM sizes with some exceptions. Reservation discounts don't apply for the following VMs:

- **VM series** - A-series, Av2-series, or G-series.
- **Preview or Promo VMs** - Any VM-series or size that is in preview or uses promotional meter.
- **Clouds** - Reservations aren't available for purchase in Germany or China regions.
- **Insufficient quota** - A reservation that is scoped to a single subscription must have vCPU quota available in the subscription for the new RI. For example, if the target subscription has a quota limit of 10 vCPUs for D-Series, then you can't buy a reservation for 11 Standard\_D1 instances. The quota check for reservations includes the VMs already deployed in the subscription. For example, if the subscription has a quota of 10 vCPUs for D-Series and has two standard\_D1 instances deployed, then you can buy a reservation for 10 standard\_D1 instances in this subscription. You can [create quote increase request](#) to resolve this issue.
- **Capacity restrictions** - In rare circumstances, Azure limits the purchase of new reservations for subset of VM sizes, because of low capacity in a region.

## Buy a Reserved VM Instance

You can buy a reserved VM instance in the [Azure portal](#). Pay for the reservation [up front or with monthly payments](#). These requirements apply to buying a reserved VM instance:

- You must be in an Owner role for at least one EA subscription or a subscription with a pay-as-you-go rate.
- For EA subscriptions, the **Add Reserved Instances** option must be enabled in the [EA portal](#). Or, if that setting is disabled, you must be an EA Admin for the subscription.
- For the Cloud Solution Provider (CSP) program, only the admin agents or sales agents can buy reservations.

To buy an instance:

1. Sign in to the [Azure portal](#).
2. Select **All services > Reservations**.
3. Select **Add** to purchase a new reservation and then click **Virtual machine**.
4. Enter required fields. Running VM instances that match the attributes you select qualify to get the reservation discount. The actual number of your VM instances that get the discount depend on the scope and quantity selected.

If you have an EA agreement, you can use the **Add more option** to quickly add additional instances. The option isn't available for other subscription types.

FIELD	DESCRIPTION
Subscription	The subscription used to pay for the reservation. The payment method on the subscription is charged the costs for the reservation. The subscription type must be an enterprise agreement (offer numbers: MS-AZR-0017P or MS-AZR-0148P) or Microsoft Customer Agreement or an individual subscription with pay-as-you-go rates (offer numbers: MS-AZR-0003P or MS-AZR-0023P). The charges are deducted from the Azure Prepayment (previously called monetary commitment) balance, if available, or charged as overage. For a subscription with pay-as-you-go rates, the charges are billed to the credit card or invoice payment method on the subscription.
Scope	The reservation's scope can cover one subscription or multiple subscriptions (shared scope). If you select: <ul style="list-style-type: none"><li>• <b>Single resource group scope</b> — Applies the reservation discount to the matching resources in the selected resource group only.</li><li>• <b>Single subscription scope</b> — Applies the reservation discount to the matching resources in the selected subscription.</li><li>• <b>Shared scope</b> — Applies the reservation discount to matching resources in eligible subscriptions that are in the billing context. For EA customers, the billing context is the enrollment. For individual subscriptions with pay-as-you-go rates, the billing scope is all eligible subscriptions created by the account administrator.</li><li>• <b>Management group</b> - Applies the reservation discount to the matching resource in the list of subscriptions that are a part of both the management group and billing scope.</li></ul>
Region	The Azure region that's covered by the reservation.
VM Size	The size of the VM instances.

FIELD	DESCRIPTION
Optimize for	VM instance size flexibility is selected by default. Click <b>Advanced settings</b> to change the instance size flexibility value to apply the reservation discount to other VMs in the same <a href="#">VM size group</a> . Capacity priority prioritizes data center capacity for your deployments. It offers additional confidence in your ability to launch the VM instances when you need them. Capacity priority is only available when the reservation scope is single subscription.
Term	One year or three years. There's also a 5-year term available only for HBv2 VMs.
Quantity	The number of instances being purchased within the reservation. The quantity is the number of running VM instances that can get the billing discount. For example, if you are running 10 Standard_D2 VMs in the East US, then you would specify quantity as 10 to maximize the benefit for all running VMs.

## Usage data and reservation utilization

Your usage data has an effective price of zero for the usage that gets a reservation discount. You can see which VM instance received the reservation discount for each reservation.

For more information about how reservation discounts appear in usage data, see [Understand Azure reservation usage for your Enterprise enrollment](#) if you are an EA customer. If you have an individual subscription, see [Understand Azure reservation usage for your Pay-As-You-Go subscription](#).

## Change a reservation after purchase

You can make the following types of changes to a reservation after purchase:

- Update reservation scope
- Instance size flexibility (if applicable)
- Ownership

You can also split a reservation into smaller chunks and merge already split reservations. None of the changes cause a new commercial transaction or change the end date of the reservation.

You can't make the following types of changes after purchase, directly:

- An existing reservation's region
- SKU
- Quantity
- Duration

However, you can *exchange* a reservation if you want to make changes.

## Cancel, exchange, or refund reservations

You can cancel, exchange, or refund reservations with certain limitations. For more information, see [Self-service exchanges and refunds for Azure Reservations](#).

## Need help? Contact us.

If you have questions or need help, [create a support request](#).

## Next steps

- To learn how to manage a reservation, see [Manage Azure Reservations](#).
- To learn more about Azure Reservations, see the following articles:
  - [What are Azure Reservations?](#)
  - [Manage Reservations in Azure](#)
  - [Understand how the reservation discount is applied](#)
  - [Understand reservation usage for a subscription with pay-as-you-go rates](#)
  - [Understand reservation usage for your Enterprise enrollment](#)
  - [Windows software costs not included with reservations](#)
  - [Azure Reservations in Partner Center Cloud Solution Provider \(CSP\) program](#)

# Save costs with Azure Dedicated Host reservations

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

When you commit to a reserved instance of Azure Dedicated Hosts, you can save money. The reservation discount is applied automatically to the number of running dedicated hosts that match the reservation scope and attributes. You don't need to assign a reservation to a dedicated host to get the discounts. A reserved instance purchase covers only the compute part of your usage and does include software licensing costs. See the [Overview of Azure Dedicated Hosts for virtual machines](#).

## Determine the right dedicated host SKU before you buy

Before you buy a reservation, you should determine which dedicated host you need. A SKU is defined for a dedicated host representing the VM series and type.

Start by going over the supported sizes for [Windows virtual machine](#) or [Linux](#) to identify the VM series.

Next, check whether it is supported on Azure Dedicated Hosts. [Azure Dedicated Hosts pricing](#) page has the complete list of dedicated hosts SKUs, their CPU information, and various pricing options (including reserved instances).

You may find several SKUs supporting your selected VM series (with different Types). Identify the best SKU by comparing the capacity of the host (number of vCPUs). Note that you will be able to apply your reservation to multiple dedicated hosts SKUs supporting the same VM series (for example DSv3\_Type1 and DSv3\_Type2) but not across different VM series (like DSv3 and ESv3).

## Purchase restriction considerations

Reserved instances are available for most dedicated host sizes, with some exceptions.

Reservation discounts don't apply for the following:

- **Clouds** - Reservations aren't available for purchase in Germany or China regions.
- **Insufficient quota** - A reservation that is scoped to a single subscription must have vCPU quota available in the subscription for the new reserved instance. For example, if the target subscription has a quota limit of 10 vCPUs for DSv3-Series, then you can't buy a reservation dedicated hosts supporting this series. The quota check for reservations includes the VMs and dedicated hosts already deployed in the subscription. You can [create quota increase request](#) to resolve this issue.
- **Capacity restrictions** - In rare circumstances, Azure limits the purchase of new reservations for subset of dedicated host SKUs, because of low capacity in a region.

## Buy a reservation

You can buy a reserved instance of an Azure Dedicated Host instance in the [Azure portal](#).

Pay for the reservation [up front or with monthly payments](#). These requirements apply to buying a reserved Dedicated Host instance:

- You must be in an Owner role for at least one EA subscription or a subscription with a pay-as-you-go rate.

- For EA subscriptions, the **Add Reserved Instances** option must be enabled in the [EA portal](#). Or, if that setting is disabled, you must be an EA Admin for the subscription.
- For the Cloud Solution Provider (CSP) program, only the admin agents or sales agents can buy reservations.

To buy an instance:

- Sign in to the [Azure portal](#).
- Select **All services > Reservations**.
- Select **Add** to purchase a new reservation and then click **Dedicated Hosts**.
- Enter required fields. Running Dedicated Hosts instances that match the attributes you select qualify to get the reservation discount. The actual number of your Dedicated Host instances that get the discount depend on the scope and quantity selected.

If you have an EA agreement, you can use the **Add more option** to quickly add additional instances. The option isn't available for other subscription types.

FIELD	DESCRIPTION
Subscription	The subscription used to pay for the reservation. The payment method on the subscription is charged the costs for the reservation. The subscription type must be an enterprise agreement (offer numbers: MS-AZR-0017P or MS-AZR-0148P) or Microsoft Customer Agreement or an individual subscription with pay-as-you-go rates (offer numbers: MS-AZR-0003P or MS-AZR-0023P). The charges are deducted from the Azure Prepayment (previously called monetary commitment) balance, if available, or charged as overage. For a subscription with pay-as-you-go rates, the charges are billed to the credit card or invoice payment method on the subscription.
Scope	The reservation's scope can cover one subscription or multiple subscriptions (shared scope). If you select:
Region	The Azure region that's covered by the reservation.
Dedicated Host Size	The size of the Dedicated Host instances.
Term	One year or three years.
Quantity	The number of instances being purchased within the reservation. The quantity is the number of running Dedicated Host instances that can get the billing discount.

- Single resource group scope** — Applies the reservation discount to the matching resources in the selected resource group only.
- Single subscription scope** — Applies the reservation discount to the matching resources in the selected subscription.
- Shared scope** — Applies the reservation discount to matching resources in eligible subscriptions that are in the billing context. For EA customers, the billing context is the enrollment. For individual subscriptions with pay-as-you-go rates, the billing scope is all eligible subscriptions created by the account administrator.

- **Management group** — Applies the reservation discount to the matching resource in the list of subscriptions that are a part of both the management group and billing scope.

## Usage data and reservation utilization

Your usage data has an effective price of zero for the usage that gets a reservation discount. You can see which VM instance received the reservation discount for each reservation.

For more information about how reservation discounts appear in usage data, see [Understand Azure reservation usage for your Enterprise enrollment](#) if you are an EA customer. If you have an individual subscription, see [Understand Azure reservation usage for your Pay-As-You-Go subscription](#).

## Change a reservation after purchase

You can make the following types of changes to a reservation after purchase:

- Update reservation scope
- Instance size flexibility (if applicable)
- Ownership

You can also split a reservation into smaller chunks and merge already split reservations. None of the changes cause a new commercial transaction or change the end date of the reservation.

You can't make the following types of changes after purchase, directly:

- An existing reservation's region
- SKU
- Quantity
- Duration

However, you can *exchange* a reservation if you want to make changes.

## Cancel, exchange, or refund reservations

You can cancel, exchange, or refund reservations with certain limitations. For more information, see [Self-service exchanges and refunds for Azure Reservations](#).

## Need help? Contact us.

If you have questions or need help, [create a support request](#).

## Next steps

To learn how to manage a reservation, see [Manage Azure Reservations](#).

To learn more about Azure Reservations, see the following articles:

- [What are Azure Reservations?](#)
- [Using Azure Dedicated Hosts](#)
- [Dedicated Hosts Pricing](#)
- [Manage Reservations in Azure](#)

- Understand how the reservation discount is applied
- Understand reservation usage for a subscription with pay-as-you-go rates
- Understand reservation usage for your Enterprise enrollment
- Windows software costs not included with reservations
- Azure Reservations in Partner Center Cloud Solution Provider (CSP) program

# Virtual machine size flexibility with Reserved VM Instances

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

When you buy a Reserved VM Instance, you can choose to optimize for instance size flexibility or capacity priority. For more information about setting or changing the optimize setting for reserved VM instances, see [Change the optimize setting for reserved VM instances](#).

With a reserved virtual machine instance that's optimized for instance size flexibility, the reservation you buy can apply to the virtual machines (VMs) sizes in the same instance size flexibility group. For example, if you buy a reservation for a VM size that's listed in the DSv2 Series, like Standard\_DS3\_v2, the reservation discount can apply to the other sizes that are listed in that same instance size flexibility group:

- Standard\_DS1\_v2
- Standard\_DS2\_v2
- Standard\_DS3\_v2
- Standard\_DS4\_v2

But that reservation discount doesn't apply to VMs sizes that are listed in different instance size flexibility groups, like SKUs in DSv2 Series High Memory: Standard\_DS11\_v2, Standard\_DS12\_v2, and so on.

Within the instance size flexibility group, the number of VMs the reservation discount applies to depends on the VM size you pick when you buy a reservation. It also depends on the sizes of the VMs that you have running. The ratio column compares the relative footprint for each VM size in that instance size flexibility group. Use the ratio value to calculate how the reservation discount applies to the VMs you have running.

## Examples

The following examples use the sizes and ratios in the DSv2-series table.

You buy a reserved VM instance with the size Standard\_DS4\_v2 where the ratio or relative footprint compared to the other sizes in that series is 8.

- Scenario 1: Run eight Standard\_DS1\_v2 sized VMs with a ratio of 1. Your reservation discount applies to all eight of those VMs.
- Scenario 2: Run two Standard\_DS2\_v2 sized VMs with a ratio of 2 each. Also run a Standard\_DS3\_v2 sized VM with a ratio of 4. The total footprint is  $2+2+4=8$ . So your reservation discount applies to all three of those VMs.
- Scenario 3: Run one Standard\_DS5\_v2 with a ratio of 16. Your reservation discount applies to half that VM's compute cost.
- Scenario 4: Run one Standard\_DS5\_v2 with a ratio of 16 and purchase an additional Standard\_DS4\_v2 reservation with a ratio of 8. Both reservations combine and apply the discount to entire VM.

The following sections show what sizes are in the same size series group when you buy a reserved VM instance optimized for instance size flexibility.

## Instance size flexibility ratio for VMs

CSV below has the instance size flexibility groups, ArmSkuName and the ratios.

## Instance size flexibility ratios

Azure keeps link and schema updated so that you can use the file programmatically.

## View VM size recommendations

Azure shows VM size recommendations in the purchase experience. To view the smallest size recommendations, select **Group by smallest size**.

### Select the product you want to purchase

X

Reserved VM Instances (RIs) provide a significant discount over pay-as-you-go VM prices by allowing you to pre-purchase the base costs of your VM usage for a period of 1 or 3 years. Reserved instance discount will automatically apply to matching VMs; you don't need to re-deploy resources to get reservation discount. The reservation applies only to hardware usage. Windows is charged separately. [Learn More](#)

Scope \*  Billing subscription \*

Recommended All Products

Filter by name, region, or instance flexi...									Region : <b>Select a value</b> X	Term : <b>Three Years</b> X	Billing frequency : <b>Monthly</b> X	+ Y Add Filter	Reset filters				
1-49 of 49									Recommendations based on 30 day usage Learn more								
↑↓	Name ↑↓	Region ↑↓	Instance flexibility group ↑↓	vCPUs ↑↓	RAM (GB) ↑↓	Term ↑↓	Billing freque... ↑↓	Recommended quantity ↑↓									
 Standard_DS1_v2	West US 2	DSv2 Series		1	3.5	Three Years	Monthly	<a href="#">18 - See details</a>									
 Standard_D2s_v3	Brazil Southea...	DSv3 Series		2	8	Three Years	Monthly	<a href="#">10 - See details</a>									
 Standard_DS2_v2	West Europe	DSv2 Series		2	7	Three Years	Monthly	<a href="#">9 - See details</a>									
 Standard_DS1_v2	East US 2	DSv2 Series		1	3.5	Three Years	Monthly	<a href="#">7 - See details</a>									
 Standard_D2s_v3	East US 2	DSv3 Series		2	8	Three Years	Monthly	<a href="#">5 - See details</a>									
 Standard_D4s_v3	West Europe	DSv3 Series		4	16	Three Years	Monthly	<a href="#">5 - See details</a>									
 Standard_D8s_v3	West Europe	DSv3 Series		8	32	Three Years	Monthly	<a href="#">5 - See details</a>									
 Standard_D15_v2	East US 2	Dv2 Series High Memory		20	140	Three Years	Monthly	<a href="#">5 - See details</a>									
 Standard_DS1_v2	East US	DSv2 Series		1	3.5	Three Years	Monthly	<a href="#">4 - See details</a>									
 Standard_DS2_v2	Southeast Asia	DSv2 Series		2	7	Three Years	Monthly	<a href="#">4 - See details</a>									
 Standard_DS1_v2	Canada Central	DSv2 Series		1	3.5	Three Years	Monthly	<a href="#">3 - See details</a>									
 Standard_DS2_v2	East US	DSv2 Series		2	7	Three Years	Monthly	<a href="#">3 - See details</a>									
 Standard_DS2_v2	West US 2	DSv2 Series		2	7	Three Years	Monthly	<a href="#">3 - See details</a>									
 Standard_D2s_v3	West US 2	DSv3 Series		2	8	Three Years	Monthly	<a href="#">3 - See details</a>									
 Standard_F2	Southeast Asia	F Series		2	4	Three Years	Monthly	<a href="#">3 - See details</a>									
 Standard_B2s	Japan East	BS Series		2	4	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_DS2_v2	East US 2	DSv2 Series		2	7	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_DS2_v2	North Europe	DSv2 Series		2	7	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_DS3_v2	Southeast Asia	DSv2 Series		4	14	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_DS12_v2	West US 2	DSv2 Series High Memory		4	28	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_D2s_v3	Central US	DSv3 Series		2	8	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_D2s_v3	Southeast Asia	DSv3 Series		2	8	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_D2s_v3	West Europe	DSv3 Series		2	8	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_F2s_v2	East US	FSv2 Series		2	4	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_F4s_v2	Southeast Asia	FSv2 Series		4	8	Three Years	Monthly	<a href="#">2 - See details</a>									
 Standard_B1s	Japan East	BS Series		1	0.5	Three Years	Monthly	<a href="#">1 - See details</a>									

< Previous

Page 1 of 1

Next >

Not seeing what you want? [Browse all products.](#)

Add to cart

Close

## Next steps

For more information, see [What are Azure Reservations](#).

# On-demand Capacity Reservation

9/21/2022 • 12 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale set ✓ Flexible scale sets

On-demand Capacity Reservation enables you to reserve Compute capacity in an Azure region or an Availability Zone for any duration of time. Unlike [Reserved Instances](#), you do not have to sign up for a 1-year or a 3-year term commitment. Create and delete reservations at any time and have full control over how you want to manage your reservations.

Once the Capacity Reservation is created, the capacity is available immediately and is exclusively reserved for your use until the reservation is deleted.

Capacity Reservation has some basic properties that are always defined at the time of creation:

- **VM size**- Each reservation is for one VM size. For example, `Standard_D2s_v3`.
- **Location**- Each reservation is for one location (region). If that location has availability zones, then the reservation can also specify one of the zones.
- **Quantity**- Each reservation has a quantity of instances to be reserved.

To create a Capacity Reservation, these parameters are passed to Azure as a capacity request. If the subscription lacks the required quota or Azure does not have capacity available that meets the specification, the reservation will fail to deploy. To avoid deployment failure, request more quota or try a different VM size, location, or zone combination.

Once Azure accepts a reservation request, it is available to be consumed by VMs of matching configurations. To consume Capacity Reservation, the VM will have to specify the reservation as one of its properties. Otherwise, the Capacity Reservation will remain unused. One benefit of this design is that you can target only critical workloads to reservations and other non-critical workloads can run without reserved capacity.

## Benefits of Capacity Reservation

- Once deployed, capacity is reserved for your use and always available within the scope of applicable SLAs
- Can be deployed and deleted at any time with no term commitment
- Can be combined automatically with Reserved Instances to use term commitment discounts

## SLA for Capacity Reservation

Please read the Service Level Agreement details in the [SLA for Capacity Reservation](#).

Any claim against the SLA requires calculating the Minutes Not Available for the reserved capacity. Here is an example of how to calculate Minutes Not Available.

- A On Demand Capacity Reservation has a total Capacity of 5 Reserved Units. The On Demand Capacity Reservation starts in the Unused Capacity state with 0 Virtual Machines Allocated.
- A Supported Deployment of quantity 5 is allocated to the On Demand Capacity Reservation. 3 Virtual Machines succeed and 2 fail with a Virtual Machine capacity error. Result: 2 Reserved Units begin to accumulate Minutes Not Available.
- No action is taken for 20 minutes. Result: two Reserved Units each accumulate 15 Minutes Not Available.
- At 20 minutes, a Supported Deployment of quantity 2 is attempted. One Virtual Machine succeeds, the other Virtual Machine fails with a Virtual Machine capacity error. Result: One Reserved Unit stays at 15

accumulated Minutes Not Available. Another Reserved Unit resumes accumulating Minutes Not Available.

- Four additional Supported Deployments of quantity 1 are made at 10 minute intervals. On the fourth attempt (60 minutes after the first capacity error), the Virtual Machine is deployed. Result: The last Reserved Unit adds 40 minutes of Minutes Not Available (4 attempts x 10 minutes between attempts) for a total of 55 Minutes Not Available.

From this example accumulation of Minutes Not Available, here is the calculation of Service Credit.

- One Reserved Unit accumulated 15 minutes of Downtime. The Percentage Uptime is 99.97%. This Reserved Unit does not qualify for Service Credit.
- Another Reserved Unit accumulated 55 minutes of Downtime. The Percentage Uptime is 99.87. This Reserved Unit qualifies for Service Credit of 10%.

## Limitations and restrictions

- Creating capacity reservations requires quota in the same manner as creating virtual machines.
- Creating capacity reservation is currently limited to certain VM Series and Sizes. The Compute Resource SKUs list advertises the set of supported VM Sizes.
- The following VM Series support creation of capacity reservations:
  - Av2
  - B
  - D series, v2 and newer; AMD and Intel
  - E series, all versions; AMD and Intel
  - F series, all versions
  - Lsv3 (Intel) and Lasv3 (AMD)
  - At VM deployment, Fault Domain (FD) count of up to 3 may be set as desired using Virtual Machine Scale Sets. A deployment with more than 3 FDs will fail to deploy against a Capacity Reservation.
- Support for additional VM Series isn't currently available:
  - Ls and Lsv2 series
  - M series, any version
  - NC-series, v3 and newer
  - NV-series, v2 and newer
  - ND-series
  - Hb-series
  - Hc-series
- The following deployment types are supported:
  - Single VM
  - Virtual Machine Scale Sets with Uniform Orchestration
  - Virtual Machine Scale Sets with Flexible Orchestration (preview)
- The following deployment types are not supported:
  - Spot VMs
  - Azure Dedicated Host Nodes or VMs deployed to Dedicated Hosts
  - Availability Sets
- Other deployment constraints are not supported. For example:
  - Proximity Placement Group
  - Update domains
  - Virtual Machine Scale Sets with single placement group set 'true'
  - UltraSSD storage
  - VMs resuming from hibernation

- VMs requiring vnet encryption
- Only the subscription that created the reservation can use it.
- Reservations are only available to paid Azure customers. Sponsored accounts such as Free Trial and Azure for Students are not eligible to use this feature.

## Pricing and billing

Capacity Reservations are priced at the same rate as the underlying VM size. For example, if you create a reservation for ten D2s\_v3 VMs then you will start getting billed for ten D2s\_v3 VMs, even if the reservation is not being used.

If you then deploy a D2s\_v3 VM and specify reservation property, the Capacity Reservation gets used. Once in use, you will only pay for the VM and nothing extra for the Capacity Reservation. Let's say you deploy six D2s\_v3 VMs against the previously mentioned Capacity Reservation. You will see a bill for six D2s\_v3 VMs and four unused Capacity Reservation, both charged at the same rate as a D2s\_v3 VM.

Both used and unused Capacity Reservation are eligible for Reserved Instances term commitment discounts. In the previous example, if you have Reserved Instances for two D2s\_v3 VMs in the same Azure region, the billing for two resources (either VM or unused Capacity Reservation) will be zeroed out. The remaining eight D2s\_v3 will be billed normally. The term commitment discounts could be applied on either the VM or the unused Capacity Reservation.

## Difference between On-demand Capacity Reservation and Reserved Instances

DIFFERENCES	ON-DEMAND CAPACITY RESERVATION	RESERVED INSTANCES
Term	No term commitment required. Can be created and deleted as per the customer requirement	Fixed term commitment of either one-year or three-years
Billing discount	Charged at pay-as-you-go rates for the underlying VM size*	Significant cost savings over pay-as-you-go rates
Capacity SLA	Provides capacity guarantee in the specified location (region or availability zone)	Does not provide a capacity guarantee. Customers can choose "capacity priority" to gain better access, but that option does not carry an SLA
Region vs Availability Zones	Can be deployed per region or per availability zone	Only available at regional level

\*Eligible for Reserved Instances discount if purchased separately

## Work with Capacity Reservation

Capacity Reservation is created for a specific VM size in an Azure region or an Availability Zone. All reservations are created and managed as part of a Capacity Reservation Group.

The group specifies the Azure location:

- The group sets the region in which all reservations will be created. For example, East US, North Europe, or Southeast Asia.
- The group sets the eligible zones. For example, AZ1, AZ2, AZ3 in any combination.
- If no zones are specified, Azure will select the placement for the group somewhere in the region. Each

reservation will specify the region and may not set a zone.

Each reservation in a group is for one VM size. If eligible zones were selected for the group, the reservation must be for one of the supported zones.

A group can have only one reservation per VM size per zone, or just one reservation per VM size if no zones are selected.

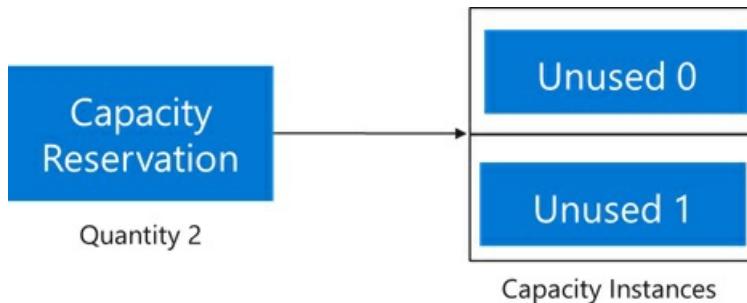
To consume Capacity Reservation, specify Capacity Reservation Group as one of the VM properties. If the group doesn't have a reservation matching the size and location, Azure will return an error message.

The quantity reserved for reservation can be adjusted after initial deployment by changing the capacity property. Other changes to Capacity Reservation, such as VM size or location, are not permitted. The recommended approach is to create a new reservation, migrate any existing VMs, and then delete the old reservation if no longer needed.

Capacity Reservation doesn't create limits on the number of VM deployments. Azure supports allocating as many VMs as desired against the reservation. As the reservation itself requires quota, the quota checks are omitted for VM deployment up to the reserved quantity. Allocating VMs beyond the reserved quantity is called overallocating the reservation. Overallocating VMs is not covered by the SLA and the VMs will be subject to quota checks and Azure fulfilling the extra capacity. Once deployed, these extra VM instances can cause the quantity of VMs allocated against the reservation to exceed the reserved quantity. To learn more, go to [Overallocating Capacity Reservation](#).

## Capacity Reservation lifecycle

When a reservation is created, Azure sets aside the requested number of capacity instances in the specified location:

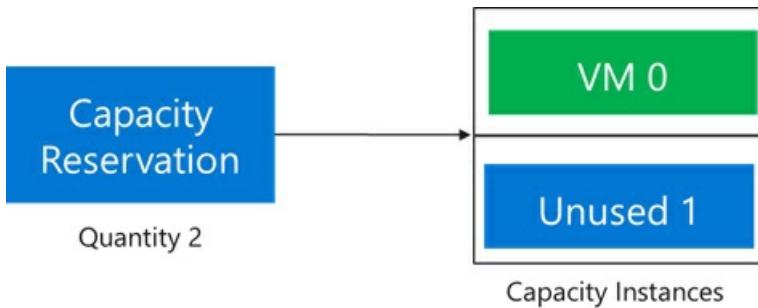


Track the state of the overall reservation through the following properties:

- `capacity` = Total quantity of instances reserved by the customer.
- `virtualMachinesAllocated` = List of VMs allocated against the Capacity Reservation and count towards consuming the capacity. These VMs are either *Running*, *Stopped (Allocated)*, or in a transitional state such as *Starting* or *Stopping*. This list doesn't include the VMs that are in deallocated state, referred to as *Stopped (deallocated)*.
- `virtualMachinesAssociated` = List of VMs associated with the Capacity Reservation. This list has all the VMs that have been configured to use the reservation, including the ones that are in deallocated state.

The previous example will start with `capacity` as 2 and length of `virtualMachinesAllocated` and `virtualMachinesAssociated` as 0.

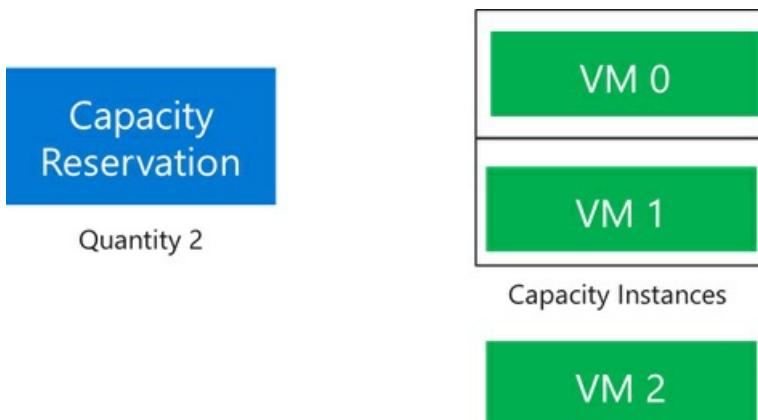
When a VM is then allocated against the Capacity Reservation, it will logically consume one of the reserved capacity instances:



The status of the Capacity Reservation will now show `capacity` as 2 and length of `virtualMachinesAllocated` and `virtualMachinesAssociated` as 1.

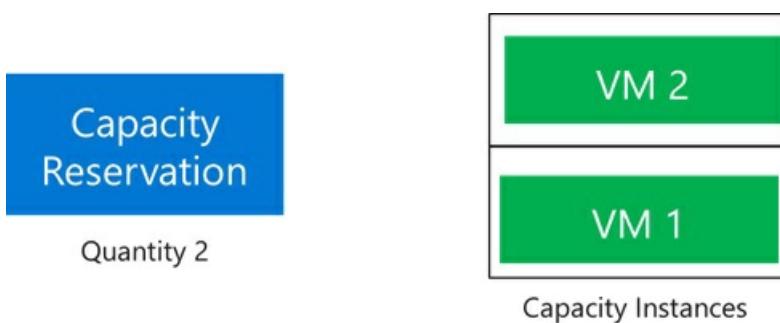
Allocations against the Capacity Reservation will succeed as long as the VMs have matching properties and there is at least one empty capacity instance.

Using our example, when a third VM is allocated against the Capacity Reservation, the reservation enters the [overallocated](#) state. This third VM will require unused quota and extra capacity fulfillment from Azure. Once the third VM is allocated, the Capacity Reservation now looks like this:



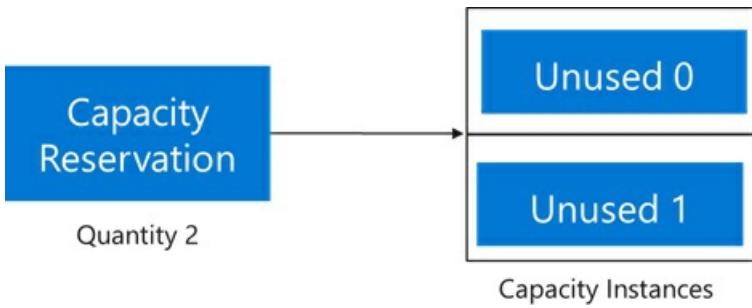
The `capacity` is 2 and the length of `virtualMachinesAllocated` and `virtualMachinesAssociated` is 3.

Now suppose the application scales down to the minimum of two VMs. Since VM 0 needs an update, it is chosen for deallocation. The reservation automatically shifts to this state:

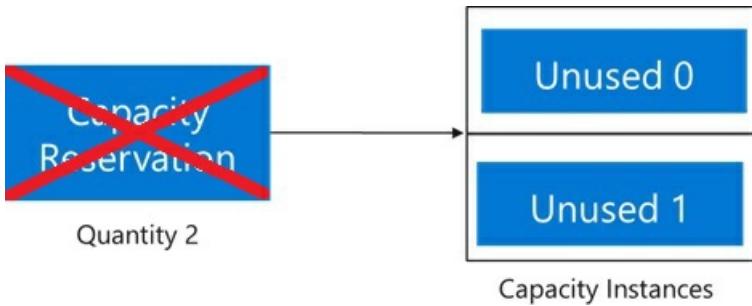


The `capacity` and the length of `virtualMachinesAllocated` are both 2. However, the length for `virtualMachinesAssociated` is still 3 as VM 0, though deallocated, is still associated with the Capacity Reservation. To prevent quota overrun, the deallocated VM 0 still counts against the quota allocated to the reservation. As long as you have enough unused quota, you can deploy new VMs to the Capacity Reservation and receive the SLA from any unused reserved capacity. Or you can delete VM 0 to remove its use of quota.

The Capacity Reservation will exist until explicitly deleted. To delete a Capacity Reservation, the first step is to dissociate all the VMs in the `virtualMachinesAssociated` property. Once disassociation is complete, the Capacity Reservation should look like this:



The status of the Capacity Reservation will now show `capacity` as 2 and length of `virtualMachinesAssociated` and `virtualMachinesAllocated` as 0. From this state, the Capacity Reservation can be deleted. Once deleted, you will not pay for the reservation anymore.

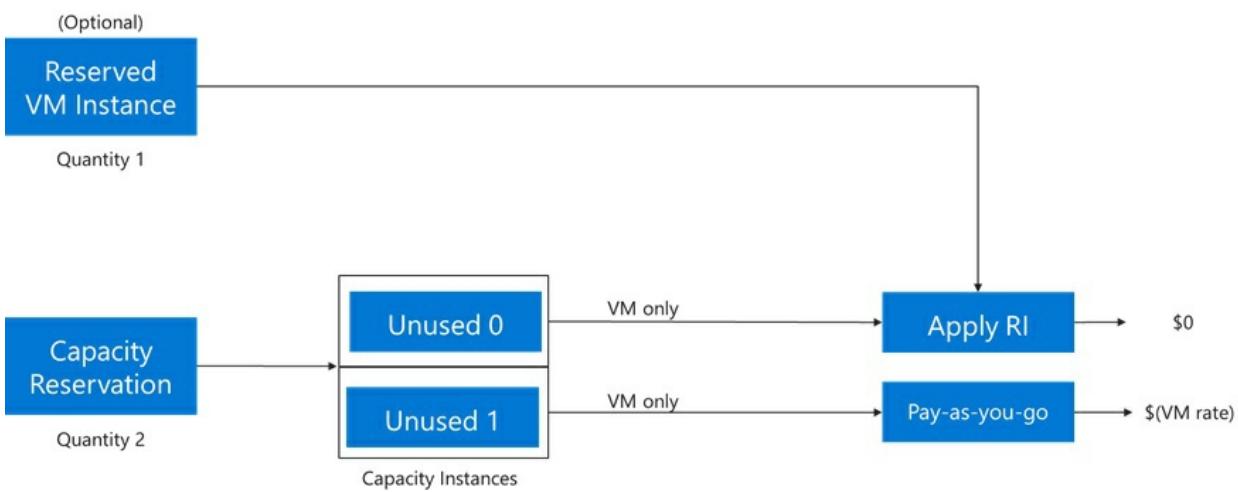


## Usage and billing

When a Capacity Reservation is empty, VM usage will be reported for the corresponding VM size and the location. [VM Reserved Instances](#) can cover some or all of the Capacity Reservation usage even when VMs are not deployed.

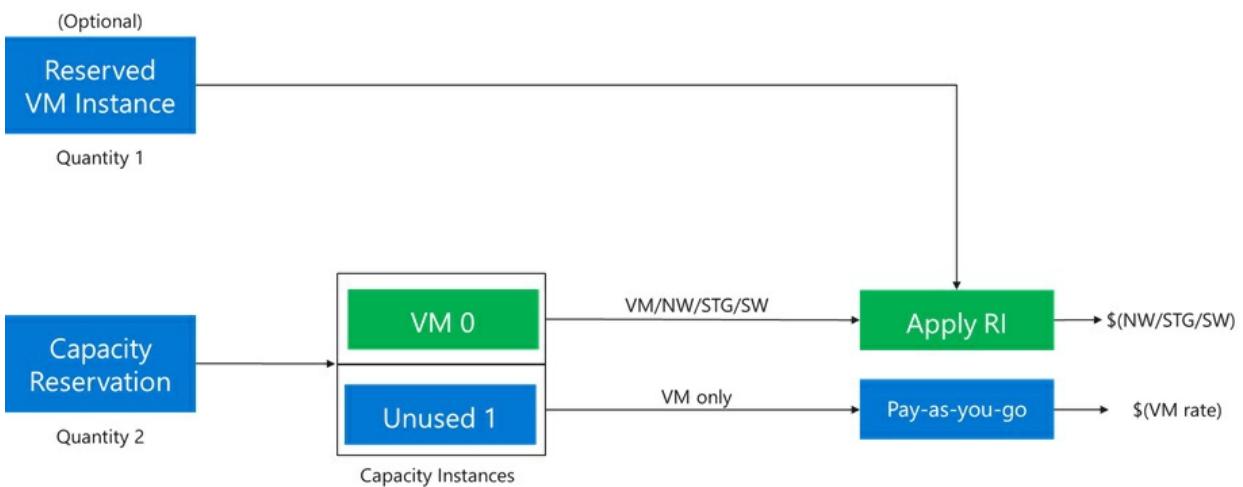
### Example

For example, let's say a Capacity Reservation with quantity reserved 2 has been created. The subscription has access to one matching Reserved VM Instance of the same size. The result is two usage streams for the Capacity Reservation, one of which is covered by the Reserved Instance:



In the previous image, a Reserved VM Instance discount is applied to one of the unused instances and the cost for that instance will be zeroed out. For the other instance, PAYG rate will be charged for the VM size reserved.

When a VM is allocated against the Capacity Reservation, the other VM components such as disks, network, extensions, and any other requested components must also be allocated. In this state, the VM usage will reflect one allocated VM and one unused capacity instance. The Reserved VM Instance will zero out the cost of either the VM or the unused capacity instance. The other charges for disks, networking, and other components associated with the allocated VM will also appear on the bill.



In the previous image, the VM Reserved Instance discount is applied to VM 0, which will only be charged for other components such as disk and networking. The other unused instance is being charged at PAYG rate for the VM size reserved.

## Frequently asked questions

- **What's the price of on-demand Capacity Reservation?**

The price of your on-demand Capacity Reservation is same as the price of underlying VM size associated with the reservation. When using Capacity Reservation, you will be charged for the VM size you selected at pay-as-you-go rates, whether the VM has been provisioned or not. Visit the [Windows](#) and [Linux](#) VM pricing pages for more details.

- **Will I get charged twice, for the cost of on-demand Capacity Reservation and for the actual VM when I finally provision it?**

No, you will only get charged once for on-demand Capacity Reservation.

- **Can I apply Reserved Virtual Machine Instance (RI) to on-demand Capacity Reservation to lower my costs?**

Yes, you can apply existing or future RIs to on-demand capacity reservations and receive RI discounts. Available RIs are applied automatically to Capacity Reservation the same way they are applied to VMs.

- **What is the difference between Reserved Virtual Machine Instance (RI) and on-demand Capacity Reservation?**

Both RIs and on-demand capacity reservations are applicable to Azure VMs. However, RIs provide discounted reservation rates for your VMs compared to pay-as-you-go rates as a result of a 1-year or 3-year term commitment. Conversely, on-demand capacity reservations do not require a commitment. You can create or cancel a Capacity Reservation at any time. However, no discounts are applied, and you will incur charges at pay-as-you-go rates after your Capacity Reservation has been successfully provisioned. Unlike RIs, which prioritize capacity but do not guarantee it, when you purchase an on-demand Capacity Reservation, Azure sets aside compute capacity for your VM and provides an SLA guarantee.

- **Which scenarios would benefit the most from on-demand capacity reservations?**

Typical scenarios include business continuity, disaster recovery, and scale-out of mission-critical applications.

## Next steps

Get started reserving Compute capacity. Check out our other related Capacity Reservation articles:

- [Create a capacity reservation](#)
- [Overallocating capacity reservation](#)
- [Modify a capacity reservation](#)
- [Associate a VM](#)
- [Remove a VM](#)
- [Associate a VM scale set - Flexible](#)
- [Associate a VM scale set - Uniform](#)
- [Remove a VM scale set](#)

# Create a Capacity Reservation

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale set ✓ Flexible scale sets

Capacity Reservation is always created as part of a Capacity Reservation group. The first step is to create a group if a suitable one doesn't exist already, then create reservations. Once successfully created, reservations are immediately available for use with virtual machines. The capacity is reserved for your use as long as the reservation is not deleted.

A well-formed request for Capacity Reservation group should always succeed as it does not reserve any capacity. It just acts as a container for reservations. However, a request for Capacity Reservation could fail if you do not have the required quota for the VM series or if Azure doesn't have enough capacity to fulfill the request. Either request more quota or try a different VM size, location, or zone combination.

A Capacity Reservation creation succeeds or fails in its entirety. For a request to reserve 10 instances, success is returned only if all 10 could be allocated. Otherwise, the Capacity Reservation creation will fail.

## Considerations

The Capacity Reservation must meet the following rules:

- The location parameter must match the location property for the parent Capacity Reservation group. A mismatch will result in an error.
- The VM size must be available in the target region. Otherwise, the reservation creation will fail.
- The subscription must have available quota equal to or more than the quantity of VMs being reserved for the VM series and for the region overall. If needed, [request more quota](#).
  - As needed to satisfy existing quota limits, single VMs can be done in stages. Create a capacity reservation with a smaller quantity and reallocate that quantity of virtual machines. This will free up quota to increase the quantity reserved and add more virtual machines. Alternatively, if the subscription uses different VM sizes in the same series, reserve and redeploy VMs for the first size. Then add a reservation to the group for another size and redeploy the VMs for the new size to the reservation group. Repeat until complete.
  - For Scale Sets, available quota will be required unless the Scale Set or its VM instances are deleted, capacity is reserved, and the Scale Set instances are added using reserved capacity. If the Scale Set is updated using blue green deployment, then reserve the capacity and deploy the new Scale Set to the reserved capacity at the next update.
- Each Capacity Reservation group can have exactly one reservation for a given VM size. For example, only one Capacity Reservation can be created for the VM size `Standard_D2s_v3`. Attempt to create a second reservation for `Standard_D2s_v3` in the same Capacity Reservation group will result in an error. However, another reservation can be created in the same group for other VM sizes, such as `Standard_D4s_v3`, `Standard_D8s_v3`, and so on.
- For a Capacity Reservation group that supports zones, each reservation type is defined by the combination of **VM size** and **zone**. For example, one Capacity Reservation for `Standard_D2s_v3` in `Zone 1`, another Capacity Reservation for `Standard_D2s_v3` in `Zone 2`, and a third Capacity Reservation for `Standard_D2s_v3` in `Zone 3` is supported.

## Create a Capacity Reservation

- [API](#)

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [ARM template](#)

## 1. Create a Capacity Reservation group

To create a Capacity Reservation group, construct the following PUT request on *Microsoft.Compute* provider:

```
PUT
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}&api-version=2021-04-01
```

In the request body, include the following parameter:

```
{
  "location": "eastus"
}
```

This group is created to contain reservations for the US East location.

The group in the following example will only support regional reservations, because zones were not specified at the time of creation. To create a zonal group, pass an extra parameter *zones* in the request body:

```
{
  "location": "eastus",
  "zones": ["1", "2", "3"]
}
```

## 2. Create a Capacity Reservation

To create a reservation, construct the following PUT request on *Microsoft.Compute* provider:

```
PUT
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName}&api-version=2021-04-01
```

In the request body, include the following parameters:

```
{
  "location": "eastus",
  "sku": {
    "name": "Standard_D2s_v3",
    "capacity": 5
  },
  "tags": {
    "environment": "testing"
  }
}
```

The above request creates a reservation in the East US location for 5 quantities of the D2s\_v3 VM size.

## Check on your Capacity Reservation

Once successfully created, the Capacity Reservation is immediately available for use with VMs.

- [API](#)
- [CLI](#)
- [PowerShell](#)
- [Portal](#)

```
GET
```

```
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName}?api-version=2021-04-01
```

```
{
  "name": "<CapacityReservationName>",
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{CapacityReservationName}",
  "type": "Microsoft.Compute/capacityReservationGroups/capacityReservations",
  "location": "eastus",
  "tags": {
    "environment": "testing"
  },
  "sku": {
    "name": "Standard_D2s_v3",
    "capacity": 5
  },
  "properties": {
    "reservationId": "<reservationId>",
    "provisioningTime": "<provisioningTime>",
    "provisioningState": "Updating"
  }
}
```

## Next steps

[Learn how to modify your Capacity Reservation](#)

# Overallocating Capacity Reservation

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale set ✓ Flexible scale sets

Azure permits association of extra VMs beyond the reserved count of a Capacity Reservation to facilitate burst and other scale-out scenarios, without the overhead of managing around the limits of reserved capacity. The only difference is that the count of VMs beyond the quantity reserved does not receive the capacity availability SLA benefit. As long as Azure has available capacity that meets the virtual machine requirements, the extra allocations will succeed.

The Instance View of a Capacity Reservation group provides a snapshot of usage for each member Capacity Reservation. You can use the Instance View to see how overallocation works.

This article assumes you have created a Capacity Reservation group (`myCapacityReservationGroup`), a member Capacity Reservation (`myCapacityReservation`), and a virtual machine (`myVM1`) that is associated to the group. Go to [Create a Capacity Reservation](#) and [Associate a VM to a Capacity Reservation](#) for more details.

## Instance View for Capacity Reservation group

The Instance View for a Capacity Reservation group will look like this:

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/myCapacityReservationGroup?$expand=instanceview&api-version=2021-04-01
```

```
{
  "name": "myCapacityReservationGroup",
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/myCapacityReservationGroup",
  "type": "Microsoft.Compute/capacityReservationGroups",
  "location": "eastus",
  "properties": {
    "capacityReservations": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/MYCAPACITYRESERVATIONGROUP/capacityReservations/MYCAPACITYRESERVATION"
      }
    ],
    "virtualMachinesAssociated": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM1"
      }
    ],
    "instanceView": {
      "capacityReservations": [
        {
          "name": "myCapacityReservation",
          "utilizationInfo": {
            "virtualMachinesAllocated": [
              {
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM1"
              }
            ]
          },
          "statuses": [
            {
              "code": "ProvisioningState/succeeded",
              "level": "Info",
              "displayStatus": "Provisioningsucceeded",
              "time": "<time>"
            }
          ]
        }
      ]
    }
  }
}
```

Let's say we create another virtual machine named *myVM2* and associate it with the above Capacity Reservation group.

The Instance View for the Capacity Reservation group will now look like this:

```
{
  "name": "myCapacityReservationGroup",
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/myCapacityReservationGroup",
  "type": "Microsoft.Compute/capacityReservationGroups",
  "location": "eastus",
  "properties": {
    "capacityReservations": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/MYCAPACITYRESERVATIONGROUP/capacityReservations/MYCAPACITYRESERVATION"
      }
    ],
    "virtualMachinesAssociated": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM1"
      },
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM2"
      }
    ],
    "instanceView": {
      "capacityReservations": [
        {
          "name": "myCapacityReservation",
          "utilizationInfo": {
            "virtualMachinesAllocated": [
              {
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM1"
              },
              {
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/myVM2"
              }
            ]
          }
        },
        {
          "statuses": [
            {
              "code": "ProvisioningState/succeeded",
              "level": "Info",
              "displayStatus": "Provisioningsucceeded",
              "time": "<time>"
            }
          ]
        }
      ]
    }
  }
}
```

Notice that the length of `virtualMachinesAllocated` (2) is greater than `capacity` (1). This valid state is referred to as *overallocated*.

#### **IMPORTANT**

Azure will not stop allocations just because a Capacity Reservation is fully consumed. Auto-scale rules, temporary scale-out, and related requirements will work beyond the quantity of reserved capacity as long as Azure has available capacity and other constraints such as available quota are met.

## States and considerations

There are three valid states for a given Capacity Reservations:

STATE	STATUS	CONSIDERATIONS
Reserved capacity available	Length of <code>virtualMachinesAllocated &lt; capacity</code>	Is all the reserved capacity needed? Optionally reduce the capacity to reduce costs.
Reservation consumed	Length of <code>virtualMachinesAllocated == capacity</code>	Additional VMs will not receive the capacity SLA unless some existing VMs are deallocated. Optionally try to increase the capacity so extra planned VMs will receive an SLA.
Reservation overallocated	Length of <code>virtualMachinesAllocated &gt; capacity</code>	Additional VMs will not receive the capacity SLA. Also, the quantity of VMs (Length of <code>virtualMachinesAllocated - capacity</code> ) will not receive a capacity SLA if deallocated. Optionally increase the capacity to add capacity SLA to more of the existing VMs.

## Next steps

[Learn how to remove VMs from a Capacity Reservation](#)

# Modify a Capacity Reservation (preview)

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Uniform scale set ✓ Flexible scale sets

After creating a Capacity Reservation group and Capacity Reservation, you may want to modify your reservations. This article explains how to do the following actions using API, Azure portal, and PowerShell.

- Update the number of instances reserved in a Capacity Reservation
- Resize VMs associated with a Capacity Reservation group
- Delete the Capacity Reservation group and Capacity Reservation

## Update the number of instances reserved

Update the number of virtual machine instances reserved in a Capacity Reservation.

### IMPORTANT

In rare cases when Azure cannot fulfill the request to increase the quantity reserved for existing Capacity Reservations, it is possible that a reservation goes into a *Failed* state and becomes unavailable until the [quantity is restored to the original amount](#).

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)

PATCH

```
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName}?api-version=2021-04-01
```

In the request body, update the `capacity` property to the new count that you want to reserve:

```
{  
  "sku":  
  {  
    "capacity": 5  
  }  
}
```

Note that the `capacity` property is set to 5 now in this example.

## Resize VMs associated with a Capacity Reservation group

You must do one of the following options if the VM being resized is currently attached to a Capacity Reservation group and that group doesn't have a reservation for the target size:

- Create a new reservation for that size
- Remove the virtual machine from the reservation group before resizing.

Check if the target size is part of the reservation group:

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)

1. Get the names of all Capacity Reservations within the group.

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}?api-version=2021-04-01
```

```
{  
  "name": "<CapacityReservationGroupName>",  
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}",  
  "type": "Microsoft.Compute/capacityReservationGroups",  
  "location": "eastUS",  
  "zones": [  
    "1"  
  ],  
  "properties": {  
    "capacityReservations": [  
      {  
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName1}"  
      },  
      {  
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName2}"  
      }  
    ]  
  }  
}
```

2. Find out the VM size reserved for each reservation. The following example is for

`capacityReservationName1`, but you can repeat this step for other reservations.

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName1}?api-version=2021-04-01
```

```
{  
  "name": "capacityReservationName1",  
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/  
capacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationNam  
e1}",  
  "type": "Microsoft.Compute/capacityReservationGroups/capacityReservations",  
  "location": "eastUS",  
  "sku": {  
    "name": "Standard_D2s_v3",  
    "capacity": 3  
  },  
  "zones": [  
    "1"  
  ],  
  "properties": {  
    "reservationId": "<reservationId>",  
    "provisioningTime": "<provisioningTime>",  
    "provisioningState": "Succeeded"  
  }  
}
```

3. Consider the following scenarios:

- If the target VM size is not part of the group, [create a new Capacity Reservation](#) for the target VM
- If the target VM size already exists in the group, [resize the virtual machine](#)

## Delete a Capacity Reservation group and Capacity Reservation

Azure allows a group to be deleted when all the member Capacity Reservations have been deleted and no VMs are associated to the group.

To delete a Capacity Reservation, first find out all of the virtual machines that are associated to it. The list of virtual machines is available under `virtualMachinesAssociated` property.

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)

First, find all virtual machines associated with the Capacity Reservation group and dissociate them.

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}?\$expand=instanceView&api-version=2021-04-01
```

```
{
  "name": "<capacityReservationGroupName>",
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{capacityReservationGroupName}",
  "type": "Microsoft.Compute/capacityReservationGroups",
  "location": "eastus",
  "properties": {
    "capacityReservations": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{capacityReservationGroupName}/capacityReservations/{capacityReservationName}"
      }
    ],
    "virtualMachinesAssociated": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName1}"
      },
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName2}"
      }
    ],
    "instanceView": {
      "capacityReservations": [
        {
          "name": "{capacityReservationName}",
          "utilizationInfo": {
            "virtualMachinesAllocated": [
              {
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName1}"
              }
            ]
          }
        },
        "statuses": [
          {
            "code": "ProvisioningState/succeeded",
            "level": "Info",
            "displayStatus": "Provisioningsucceeded",
            "time": "<time>"
          }
        ]
      }
    }
  }
}
```

From the above response, find the names of all virtual machines under the `virtualMachinesAssociated` property and remove them from the Capacity Reservation group using the steps in [Remove a VM association to a Capacity Reservation](#).

Once all the virtual machines are removed from the Capacity Reservation group, delete the member Capacity Reservation(s):

```
DELETE
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{capacityReservationName}?api-version=2021-04-01
```

Lastly, delete the parent Capacity Reservation group.

DELETE

<https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}?api-version=2021-04-01>

## Restore instance quantity

A well-formed request for reducing the quantity reserved should always succeed no matter the number of VMs associated with the reservation. However, increasing the quantity reserved may require more quota and for Azure to fulfill the additional capacity request. In a rare scenario in which Azure can't fulfill the request to increase the quantity reserved for existing reservations, it is possible that the reservation goes into a *Failed* state and becomes unavailable until the quantity reserved is restored to the original amount.

### NOTE

If a reservation is in a *Failed* state, all the VMs that are associated with the reservation will continue to work as normal.

For example, let's say `myCapacityReservation` has a quantity reserved 5. You request 5 extra instances, making the total quantity reserved equal 10. However, because of a constrained capacity situation in the region, Azure can't fulfill the additional 5 quantity requested. In this case, `myCapacityReservation` will fail to meet its intended state of 10 quantity reserved and will go into a *Failed* state.

To resolve this failure, take the following steps to locate the old quantity reserved value:

1. Go to [Application Change Analysis](#) in the Azure portal
2. Select the applicable **Subscription**, **Resource group**, and **Time range** in the filters
  - You can only go back up to 14 days in the past in the **Time range** filter
3. Search for the name of the Capacity Reservation
4. Look for the change in `sku.capacity` property for that reservation
  - The old quantity reserved will be the value under the **Old Value** column

Update `myCapacityReservation` to the old quantity reserved. Once updated, the reservation will be available immediately for use with your virtual machines.

## Next steps

[Learn how to remove VMs from a Capacity Reservation](#)

# Associate a VM to a Capacity Reservation group

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows Virtual Machines ✓ Linux Virtual Machines

Capacity reservation groups can be used with new or existing virtual machines. To learn more about Capacity Reservations, see the [overview article](#).

## Associate a new VM

To associate a new VM to the Capacity Reservation group, the group must be explicitly referenced as a property of the virtual machine. This reference protects the matching reservation in the group for applications and workloads intended to use it.

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [ARM template](#)

To add the `capacityReservationGroup` property to a VM, construct the following PUT request to the *Microsoft.Compute* provider:

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName}?api-version=2021-04-01
```

In the request body, include the `capacityReservationGroup` property:

```
{  
  "location": "eastus",  
  "properties": {  
    "hardwareProfile": {  
      "vmSize": "Standard_D2s_v3"  
    },  
    ...  
    "CapacityReservation":{  
      "capacityReservationGroup":{  
  
        "id": "subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}"  
      }  
      "storageProfile": {  
        ...  
      },  
      "osProfile": {  
        ...  
      },  
      "networkProfile": {  
        ...  
      }  
    }  
  }  
}
```

## Associate an existing VM

For the initial release of Capacity Reservation, a virtual machine must be allocated to a capacity reservation.

- If not already complete, follow guidance to create a capacity reservation group and capacity reservation. Or increment the quantity of an existing capacity reservation so there's unused reserved capacity.
  - Deallocate the VM.
  - Update the capacity reservation group property on the VM.
  - Restart the VM.
- 
- [API](#)
  - [Portal](#)
  - [CLI](#)
  - [PowerShell](#)

1. Deallocate the VM.

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName}/deallocate?api-version=2021-04-01
```

2. Add the `capacityReservationGroup` property to the VM. Construct the following PUT request to *Microsoft.Compute* provider:

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName}?api-version=2021-04-01
```

In the request body, include the `capacityReservationGroup` property:

```
{  
    "location": "eastus",  
    "properties": {  
        "capacityReservation": {  
            "capacityReservationGroup": {  
                "id":  
                    "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{capacityReservationGroupName}"  
            }  
        }  
    }  
}
```

## View VM association with Instance View

Once the `capacityReservationGroup` property is set, an association now exists between the VM and the group. Azure automatically finds the matching Capacity Reservation in the group and consumes a reserved slot. The Capacity Reservation's *Instance View* will reflect the new VM in the `virtualMachinesAllocated` property:

- [API](#)
- [CLI](#)
- [PowerShell](#)
- [Portal](#)

GET

[https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{capacityReservationGroupName}?\\$expand=instanceView&api-version=2021-04-01](https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{capacityReservationGroupName}?$expand=instanceView&api-version=2021-04-01)

```
{  
  "name": "{CapacityReservationGroupName}",  
  
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}",  
  "type": "Microsoft.Compute/capacityReservationGroups",  
  "location": "eastus",  
  "properties": {  
    "capacityReservations": [  
      {  
  
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/ {CapacityReservationGroupName}/capacityReservations/{CapacityReservationName}"  
      }  
    ],  
    "virtualMachinesAssociated": [  
      {  
  
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{myVM}"  
      }  
    ],  
    "instanceView": {  
      "capacityReservations": [  
        {  
          "name": "{CapacityReservationName}",  
          "utilizationInfo": {  
            "virtualMachinesAllocated": [  
              {  
  
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{myVM}"  
              }  
            ]  
          },  
          "statuses": [  
            {  
              "code": "ProvisioningState/succeeded",  
              "level": "Info",  
              "displayStatus": "Provisioning succeeded",  
              "time": "2021-05-25T15:12:10.4165243+00:00"  
            }  
          ]  
        }  
      ]  
    }  
  }  
}
```

## Next steps

[Remove a VMs association to a Capacity Reservation group](#)

# Remove a VM association from a Capacity Reservation group

9/21/2022 • 3 minutes to read • [Edit Online](#)

This article walks you through the steps of removing a VM association to a Capacity Reservation group. To learn more about capacity reservations, see the [overview article](#).

Because both the VM and the underlying Capacity Reservation logically occupy capacity, Azure imposes some constraints on this process to avoid ambiguous allocation states and unexpected errors.

There are two ways to change an association:

- Option 1: Deallocate the virtual machine, change the Capacity Reservation group property, and optionally restart the virtual machine
- Option 2: Update the reserved quantity to zero and then change the Capacity Reservation group property

## Deallocate the VM

The first option is to deallocate the VM, change the Capacity Reservation group property, and optionally restart the VM.

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)

### 1. Deallocate the VM

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{virtualMachineName}/deallocate?api-version=2021-04-01
```

### 2. Update the VM to remove association with the Capacity Reservation group

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{virtualMachineName}/update?api-version=2021-04-01
```

In the request body, set the `capacityReservationGroup` property to null to remove the VM association to the group:

```
{  
  "location": "eastus",  
  "properties": {  
    "capacityReservation": {  
      "capacityReservationGroup": {  
        "id": null  
      }  
    }  
  }  
}
```

## Update the reserved quantity to zero

The second option involves updating the reserved quantity to zero and then changing the Capacity Reservation group property.

This option works well when the virtual machine can't be deallocated and when a reservation is no longer needed. For example, you may create a Capacity Reservation to temporarily assure capacity during a large-scale deployment. Once completed, the reservation is no longer needed.

- [API](#)
- [Portal](#)
- [CLI](#)
- [PowerShell](#)

### 1. Update the reserved quantity to zero

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/CapacityReservations/{CapacityReservationName}?api-version=2021-04-01
```

In the request body, include the following parameters:

```
{  
  "sku":  
  {  
    "capacity": 0  
  }  
}
```

Note that `capacity` property is set to 0.

### 2. Update the VM to remove the association with the Capacity Reservation group

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{VirtualMachineName}/update?api-version=2021-04-01
```

In the request body, set the `capacityReservationGroup` property to null to remove the association:

```
{  
  "location": "eastus",  
  "properties": {  
    "capacityReservation": {  
      "capacityReservationGroup": {  
        "id": null  
      }  
    }  
  }  
}
```

## Next steps

[Learn how to associate a scale set to a Capacity Reservation group](#)

# Associate a virtual machine scale set with flexible orchestration to a Capacity Reservation group

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Flexible scale sets

Virtual Machine Scale Sets have two modes:

- **Uniform Orchestration Mode:** In this mode, virtual machine scale sets use a VM profile or a template to scale up to the desired capacity. While there is some ability to manage or customize individual VM instances, Uniform uses identical VM instances. These instances are exposed through the virtual machine scale sets VM APIs and are not compatible with the standard Azure IaaS VM API commands. Since the scale set performs all the actual VM operations, reservations are associated with the virtual machine scale set directly. Once the scale set is associated with the reservation, all the subsequent VM allocations will be done against the reservation.
- **Flexible Orchestration Mode:** In this mode, you get more flexibility managing the individual virtual machine scale set VM instances as they can use the standard Azure IaaS VM APIs instead of using the scale set interface. To use reservations with flexible orchestration mode, define both the virtual machine scale set property and the capacity reservation property on each virtual machine.

To learn more about these modes, go to [Virtual Machine Scale Sets Orchestration Modes](#).

This content applies to the flexible orchestration mode. For uniform orchestration mode, go to [Associate a virtual machine scale set with uniform orchestration to a Capacity Reservation group](#)

## IMPORTANT

Capacity Reservations with virtual machine set using flexible orchestration is currently in public preview. This preview version is provided without a service-level agreement, and we don't recommend it for production workloads. Certain features might not be supported or might have constrained capabilities. For more information, see [Supplemental Terms of Use for Microsoft Azure Previews](#). During the preview, always attach reserved capacity during creation of new scale sets using flexible orchestration mode. There are known issues attaching capacity reservations to existing scale sets using flexible orchestration. Microsoft will update this page as more options become enabled during preview.

## Associate a new virtual machine scale set to a Capacity Reservation group

**Option 1: Add to Virtual Machine profile** - If the Scale Set with flexible orchestration includes a VM profile, add the Capacity Reservation group property to the profile during Scale Set creation. Follow the same process used for a Scale Set using uniform orchestration. For sample code, see [Associate a virtual machine scale set with uniform orchestration to a Capacity Reservation group](#).

**Option 2: Add to the first Virtual Machine deployed** - If the Scale Set omits a VM profile, then you must add the Capacity Reservation group to the first Virtual Machine deployed using the Scale Set. Follow the same process used to associate a VM. For sample code, see [Associate a virtual machine to a Capacity Reservation group](#).

## Next steps

[Learn how to remove a scale set association from a Capacity Reservation](#)

# Associate a virtual machine scale set with uniform orchestration to a Capacity Reservation group

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Uniform scale set

Virtual Machine Scale Sets have two modes:

- **Uniform Orchestration Mode:** In this mode, virtual machine scale sets use a VM profile or a template to scale up to the desired capacity. While there is some ability to manage or customize individual VM instances, Uniform uses identical VM instances. These instances are exposed through the virtual machine scale sets VM APIs and are not compatible with the standard Azure IaaS VM API commands. Since the scale set performs all the actual VM operations, reservations are associated with the virtual machine scale set directly. Once the scale set is associated with the reservation, all the subsequent VM allocations will be done against the reservation.
- **Flexible Orchestration Mode:** In this mode, you get more flexibility managing the individual virtual machine scale set VM instances as they can use the standard Azure IaaS VM APIs instead of using the scale set interface. To use reservations with flexible orchestration mode, define both the virtual machine scale set property and the capacity reservation property on each virtual machine.

To learn more about these modes, go to [Virtual Machine Scale Sets Orchestration Modes](#).

This content applies to the uniform orchestration mode. For flexible orchestration mode, go to [Associate a virtual machine scale set with flexible orchestration to a Capacity Reservation group](#)

## Limitations of scale sets in Uniform Orchestration

- For Virtual Machine Scale Sets in Uniform orchestration to be compatible with Capacity Reservation, the `singlePlacementGroup` property must be set to *False*.
- The **Static Fixed Spreading** availability option for multi-zone Uniform scale sets is not supported with Capacity Reservation. This option requires use of 5 Fault Domains while the reservations only support up to 3 Fault Domains for general purpose sizes. The recommended approach is to use the **Max Spreading** option that spreads VMs across as many FDs as possible within each zone. If needed, configure a custom Fault Domain configuration of 3 or less.

There are some other restrictions while using Capacity Reservation. For the complete list, refer the [Capacity Reservations overview](#).

## Associate a new virtual machine scale set to a Capacity Reservation group

- [API](#)
- [CLI](#)
- [PowerShell](#)
- [ARM template](#)

To associate a new Uniform virtual machine scale set to a Capacity Reservation group, construct the following PUT request to the *Microsoft.Compute* provider:

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMscaleSetName}?api-version=2021-04-01
```

Add the `capacityReservationGroup` property in the `virtualMachineProfile` property:

```
{  
    "name": "<VMscaleSetName>",  
    "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMscaleSetName}",  
    "type": "Microsoft.Compute/virtualMachineScaleSets",  
    "location": "eastus",  
    "sku": {  
        "name": "Standard_D2s_v3",  
        "tier": "Standard",  
        "capacity": 3  
    },  
    "properties": {  
        "virtualMachineProfile": {  
            "capacityReservation": {  
                "capacityReservationGroup": {  
  
                    "id": "subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroup/{CapacityReservationGroupName}"  
                }  
            },  
            "osProfile": {  
                ...  
            },  
            "storageProfile": {  
                ...  
            },  
            "networkProfile": {  
                ...,  
                "extensionProfile": {  
                    ...  
                }  
            }  
        }  
    }  
}
```

## Associate an existing virtual machine scale set to Capacity Reservation group

To add an existing Capacity Reservation Group to an existing Uniform Scale Set:

- Stop the Scale Set to deallocate the VM instances
- Update the Scale Set to use a matching Capacity Reservation Group
- Start the Scale Set

This process ensures the placement for the Capacity Reservations and Scale Set in the region are compatible.

### Important notes on Upgrade Policies

- **Automatic Upgrade** – In this mode, the scale set VM instances are automatically associated with the Capacity Reservation group without any further action from you. When the scale set VMs are reallocated, they start consuming the reserved capacity.
- **Rolling Upgrade** – In this mode, scale set VM instances are associated with the Capacity Reservation group without any further action from you. However, they are updated in batches with an optional pause time between them. When the scale set VMs are reallocated, they start consuming the reserved capacity.
- **Manual Upgrade** – In this mode, nothing happens to the scale set VM instances when the virtual machine

scale set is attached to a Capacity Reservation group. You will need to update to each scale set VM by upgrading it with the latest Scale Set model.

- [API](#)
- [CLI](#)
- [PowerShell](#)

#### 1. Deallocate the virtual machine scale set.

```
POST  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSScaleSetName}/deallocate?api-version=2021-04-01
```

#### 2. Add the `capacityReservationGroup` property to the scale set model. Construct the following PUT request to *Microsoft.Compute* provider:

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSScaleSetName}?api-version=2021-04-01
```

In the request body, include the `capacityReservationGroup` property:

```
"location": "eastus",  
"properties": {  
    "virtualMachineProfile": {  
        "capacityReservation": {  
            "capacityReservationGroup": {  
                "id":  
                    "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{capacityReservationGroupName}"  
            }  
        }  
    }  
}
```

## View virtual machine scale set association with Instance View

Once the Uniform virtual machine scale set is associated with the Capacity Reservation group, all the subsequent VM allocations will happen against the Capacity Reservation. Azure automatically finds the matching Capacity Reservation in the group and consumes a reserved slot.

- [API](#)
- [CLI](#)
- [PowerShell](#)
- [Portal](#)

The Capacity Reservation group *Instance View* will reflect the new scale set VMs under the `virtualMachinesAssociated` & `virtualMachinesAllocated` properties:

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}?$expand=instanceview&api-version=2021-04-01
```

```
{
  "name": "<CapacityReservationGroupName>",
  "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}",
  "type": "Microsoft.Compute/capacityReservationGroups",
  "location": "eastus"
},
  "properties": {
    "capacityReservations": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/capacityReservationGroups/{CapacityReservationGroupName}/capacityReservations/{CapacityReservationName}"
      }
    ],
    "virtualMachinesAssociated": [
      {
        "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSScaleSetName}/virtualMachines/{VirtualMachineId}"
      }
    ],
    "instanceView": {
      "capacityReservations": [
        {
          "name": "<CapacityReservationName>",
          "utilizationInfo": {
            "virtualMachinesAllocated": [
              {
                "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSScaleSetName}/virtualMachines/{VirtualMachineId}"
              }
            ]
          },
          "statuses": [
            {
              "code": "ProvisioningState/succeeded",
              "level": "Info",
              "displayStatus": "Provisioningsucceeded",
              "time": "2021-05-25T15:12:10.4165243+00:00"
            }
          ]
        }
      ]
    }
  }
}
```

## Region and Availability Zones considerations

Virtual machine scale sets can be created regionally or in one or more Availability Zones to protect them from data-center-level failure. Learn more about multi-zonal virtual machine scale sets, refer to [Virtual Machine Scale Sets that use Availability Zones](#).

### IMPORTANT

The location (Region and Availability Zones) of the virtual machine scale set and the Capacity Reservation group must match for the association to succeed. For a regional scale set, the region must match between the scale set and the Capacity Reservation group. For a zonal scale set, both the regions and the zones must match between the scale set and the Capacity Reservation group.

When a scale set is spread across multiple zones, it always attempts to deploy evenly across the included Availability Zones. Because of that even deployment, a Capacity Reservation group should always have the same

quantity of reserved VMs in each zone. As an illustration of why this is important, consider the following example.

In this example, each zone has a different quantity reserved. Let's say that the virtual machine scale set scales out to 75 instances. Since scale set will always attempt to deploy evenly across zones, the VM distribution should look like this:

ZONE	QUANTITY RESERVED	NO. OF SCALE SET VMS IN EACH ZONE	UNUSED QUANTITY RESERVED	OVERALLOCATED
1	40	25	15	0
2	20	25	0	5
3	15	25	0	10

In this case, the scale set is incurring extra cost for 15 unused instances in Zone 1. The scale-out is also relying on 5 VMs in Zone 2 and 10 VMs in Zone 3 that are not protected by Capacity Reservation. If each zone had 25 capacity instances reserved, then all 75 VMs would be protected by Capacity Reservation and the deployment would not incur any extra cost for unused instances.

Since the reservations can be overallocated, the scale set can continue to scale normally beyond the limits of the reservation. The only difference is that the VMs allocated above the quantity reserved are not covered by Capacity Reservation SLA. To learn more, go to [Overallocating Capacity Reservation](#).

## Next steps

[Learn how to remove a scale set association from a Capacity Reservation](#)

# Remove a virtual machine scale set association from a Capacity Reservation group

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Uniform scale set ✓ Flexible scale sets

This article walks you through removing a virtual machine scale set association from a Capacity Reservation group. To learn more about capacity reservations, see the [overview article](#).

Because both the VM and the underlying Capacity Reservation logically occupy capacity, Azure imposes some constraints on this process to avoid ambiguous allocation states and unexpected errors.

There are two ways to change an association:

- Option 1: Deallocate the Virtual machine scale set, change the Capacity Reservation group property at the scale set level, and then update the underlying VMs
- Option 2: Update the reserved quantity to zero and then change the Capacity Reservation group property

## Deallocate the Virtual machine scale set

The first option is to deallocate the virtual machine scale set, change the Capacity Reservation group property at the scale set level, and then update the underlying VMs.

Go to [upgrade policies](#) for more information about automatic, rolling, and manual upgrades.

- [API](#)
- [CLI](#)
- [PowerShell](#)

### 1. Deallocate the virtual machine scale set

```
POST  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMScaleSetName}/deallocate?api-version=2021-04-01
```

### 2. Update the virtual machine scale set to remove association with the Capacity Reservation group

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMScaleSetName}/update?api-version=2021-04-01
```

In the request body, set the `capacityReservationGroup` property to null to remove the virtual machine scale set association to the group:

```
{  
  "location": "eastus",  
  "properties": {  
    "virtualMachineProfile": {  
      "capacityReservation": {  
        "capacityReservationGroup": {  
          "id": null  
        }  
      }  
    }  
  }  
}
```

## Update the reserved quantity to zero

The second option involves updating the reserved quantity to zero and then changing the Capacity Reservation group property.

This option works well when the scale set cannot be deallocated and when a reservation is no longer needed. For example, you may create a Capacity Reservation to temporarily assure capacity during a large-scale deployment. Once completed, the reservation is no longer needed.

Go to [upgrade policies](#) for more information about automatic, rolling, and manual upgrades.

- [API](#)
- [CLI](#)
- [PowerShell](#)

### 1. Update the reserved quantity to zero

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/CapacityReservationGroups/{CapacityReservationGroupName}/CapacityReservations/{CapacityReservationName}?api-version=2021-04-01
```

In the request body, include the following parameters:

```
{  
  "sku":  
  {  
    "capacity": 0  
  }  
}
```

Note that `capacity` property is set to 0.

### 2. Update the virtual machine scale set to remove the association with the Capacity Reservation group

```
PUT  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSScaleSetName}/update?api-version=2021-04-01
```

In the request body, set the `capacityReservationGroup` property to null to remove the association:

```
{  
  "location": "eastus",  
  "properties": {  
    "virtualMachineProfile": {  
      "capacityReservation": {  
        "capacityReservationGroup": {  
          "id": null  
        }  
      }  
    }  
  }  
}
```

## Upgrade policies

- **Automatic Upgrade** – In this mode, the scale set VM instances are automatically dissociated from the Capacity Reservation group without any further action from you.
- **Rolling Upgrade** – In this mode, the scale set VM instances are dissociated from the Capacity Reservation group without any further action from you. However, they are updated in batches with an optional pause time between them.
- **Manual Upgrade** – In this mode, nothing happens to the scale set VM instances when the virtual machine scale set is updated. You will need to individually remove each scale set VM by [upgrading it with the latest Scale Set model](#).

## Next steps

[Learn about overallocating a Capacity Reservation](#)

# Create a complete Linux virtual machine with the Azure CLI

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

To quickly create a virtual machine (VM) in Azure, you can use a single Azure CLI command that uses default values to create any required supporting resources. Resources such as a virtual network, public IP address, and network security group rules are automatically created. For more control of your environment in production use, you may create these resources ahead of time and then add your VMs to them. This article guides you through how to create a VM and each of the supporting resources one by one.

Make sure that you have installed the latest [Azure CLI](#) and logged to an Azure account in with [az login](#).

In the following examples, replace example parameter names with your own values. Example parameter names include *myResourceGroup*, *myVnet*, and *myVM*.

## Create resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine and supporting virtual network resources. Create the resource group with [az group create](#). The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name myResourceGroup --location eastus
```

By default, the output of Azure CLI commands is in JSON (JavaScript Object Notation). To change the default output to a list or table, for example, use [az config set core.output=table](#). You can also add `--output` to any command for a one time change in output format. The following example shows the JSON output from the `az group create` command:

```
{
  "id": "/subscriptions/guid/resourceGroups/myResourceGroup",
  "location": "eastus",
  "name": "myResourceGroup",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null
}
```

## Create a virtual network and subnet

Next you create a virtual network in Azure and a subnet in to which you can create your VMs. Use [az network vnet create](#) to create a virtual network named *myVnet* with the *192.168.0.0/16* address prefix. You also add a subnet named *mySubnet* with the address prefix of *192.168.1.0/24*:

```
az network vnet create \
--resource-group myResourceGroup \
--name myVnet \
--address-prefix 192.168.0.0/16 \
--subnet-name mySubnet \
--subnet-prefix 192.168.1.0/24
```

The output shows the subnet is logically created inside the virtual network:

```
{
  "addressSpace": {
    "addressPrefixes": [
      "192.168.0.0/16"
    ]
  },
  "dhcpOptions": {
    "dnsServers": []
  },
  "etag": "W/\"e95496fc-f417-426e-a4d8-c9e4d27fc2ee\"",
  "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/virtualNetworks/myVnet",
  "location": "eastus",
  "name": "myVnet",
  "provisioningState": "Succeeded",
  "resourceGroup": "myResourceGroup",
  "resourceGuid": "ed62fd03-e9de-430b-84df-8a3b87cacdbb",
  "subnets": [
    {
      "addressPrefix": "192.168.1.0/24",
      "etag": "W/\"e95496fc-f417-426e-a4d8-c9e4d27fc2ee\"",
      "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/virtualNetworks/myVnet/subnets/mySubnet",
      "ipConfigurations": null,
      "name": "mySubnet",
      "networkSecurityGroup": null,
      "provisioningState": "Succeeded",
      "resourceGroup": "myResourceGroup",
      "resourceNavigationLinks": null,
      "routeTable": null
    }
  ],
  "tags": {},
  "type": "Microsoft.Network/virtualNetworks",
  "virtualNetworkPeerings": null
}
```

## Create a public IP address

Now let's create a public IP address with [az network public-ip create](#). This public IP address enables you to connect to your VMs from the Internet. Because the default address is dynamic, create a named DNS entry with the `--domain-name-label` parameter. The following example creates a public IP named *myPublicIP* with the DNS name of *mypublicdns*. Because the DNS name must be unique, provide your own unique DNS name:

```
az network public-ip create \
--resource-group myResourceGroup \
--name myPublicIP \
--dns-name mypublicdns
```

Output:

```
{  
  "publicIp": {  
    "dnsSettings": {  
      "domainNameLabel": "mypublicdns",  
      "fqdn": "mypublicdns.eastus.cloudapp.azure.com",  
      "reverseFqdn": null  
    },  
    "etag": "W/\"2632aa72-3d2d-4529-b38e-b622b4202925\"",  
    "id":  
      "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/publicIPAddresses/myPublicIP"  
    ,  
    "idleTimeoutInMinutes": 4,  
    "ipAddress": null,  
    "ipConfiguration": null,  
    "location": "eastus",  
    "name": "myPublicIP",  
    "provisioningState": "Succeeded",  
    "publicIpAddressVersion": "IPv4",  
    "publicIpAllocationMethod": "Dynamic",  
    "resourceGroup": "myResourceGroup",  
    "resourceGuid": "4c65de38-71f5-4684-be10-75e605b3e41f",  
    "tags": null,  
    "type": "Microsoft.Network/publicIPAddresses"  
  },  
}  
}
```

## Create a network security group

To control the flow of traffic in and out of your VMs, you apply a network security group to a virtual NIC or subnet. The following example uses [az network nsg create](#) to create a network security group named *myNetworkSecurityGroup*:

```
az network nsg create \  
  --resource-group myResourceGroup \  
  --name myNetworkSecurityGroup
```

You define rules that allow or deny specific traffic. To allow inbound connections on port 22 (to enable SSH access), create an inbound rule with [az network nsg rule create](#). The following example creates a rule named *myNetworkSecurityGroupRuleSSH*:

```
az network nsg rule create \  
  --resource-group myResourceGroup \  
  --nsg-name myNetworkSecurityGroup \  
  --name myNetworkSecurityGroupRuleSSH \  
  --protocol tcp \  
  --priority 1000 \  
  --destination-port-range 22 \  
  --access allow
```

To allow inbound connections on port 80 (for web traffic), add another network security group rule. The following example creates a rule named *myNetworkSecurityGroupRuleHTTP*:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name myNetworkSecurityGroupRuleWeb \
--protocol tcp \
--priority 1001 \
--destination-port-range 80 \
--access allow
```

Examine the network security group and rules with [az network nsg show](#):

```
az network nsg show --resource-group myResourceGroup --name myNetworkSecurityGroup
```

Output:

```
{
  "defaultSecurityRules": [
    {
      "access": "Allow",
      "description": "Allow inbound traffic from all VMs in VNET",
      "destinationAddressPrefix": "VirtualNetwork",
      "destinationPortRange": "*",
      "direction": "Inbound",
      "etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\"",
      "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/AllowVnetInBound",
      "name": "AllowVnetInBound",
      "priority": 65000,
      "protocol": "*",
      "provisioningState": "Succeeded",
      "resourceGroup": "myResourceGroup",
      "sourceAddressPrefix": "VirtualNetwork",
      "sourcePortRange": "*"
    },
    {
      "access": "Allow",
      "description": "Allow inbound traffic from azure load balancer",
      "destinationAddressPrefix": "*",
      "destinationPortRange": "*",
      "direction": "Inbound",
      "etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\"",
      "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/AllowAzureLoadBalancerInBou",
      "name": "AllowAzureLoadBalancerInBound",
      "priority": 65001,
      "protocol": "*",
      "provisioningState": "Succeeded",
      "resourceGroup": "myResourceGroup",
      "sourceAddressPrefix": "AzureLoadBalancer",
      "sourcePortRange": "*"
    },
    {
      "access": "Deny",
      "description": "Deny all inbound traffic",
      "destinationAddressPrefix": "*",
      "destinationPortRange": "*",
      "direction": "Inbound",
      "etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\"",
      "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/DenyAllInBound",
      "name": "DenyAllInBound",
```

```
"priority": 65500,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "myResourceGroup",
"sourceAddressPrefix": "*",
"sourcePortRange": "*"
},
{
"access": "Allow",
"description": "Allow outbound traffic from all VMs to all VMs in VNET",
"destinationAddressPrefix": "VirtualNetwork",
"destinationPortRange": "*",
"direction": "Outbound",
"etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\",
"id":
"/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/AllowVnetOutBound",
"name": "AllowVnetOutBound",
"priority": 65000,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "myResourceGroup",
"sourceAddressPrefix": "VirtualNetwork",
"sourcePortRange": "*"
},
{
"access": "Allow",
"description": "Allow outbound traffic from all VMs to Internet",
"destinationAddressPrefix": "Internet",
"destinationPortRange": "*",
"direction": "Outbound",
"etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\",
"id":
"/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/AllowInternetOutBound",
"name": "AllowInternetOutBound",
"priority": 65001,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "myResourceGroup",
"sourceAddressPrefix": "*",
"sourcePortRange": "*"
},
{
"access": "Deny",
"description": "Deny all outbound traffic",
"destinationAddressPrefix": "*",
"destinationPortRange": "*",
"direction": "Outbound",
"etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\",
"id":
"/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/defaultSecurityRules/DenyAllOutBound",
"name": "DenyAllOutBound",
"priority": 65500,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "myResourceGroup",
"sourceAddressPrefix": "*",
"sourcePortRange": "*"
}
],
"etag": "W/\"3371b313-ea9f-4687-a336-a8ebdfd80523\",
"id":
"/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup",
"location": "eastus",
"name": "myNetworkSecurityGroup",
"networkInterfaces": null,
```

```

    "provisioningState": "Succeeded",
    "resourceGroup": "myResourceGroup",
    "resourceGuid": "47a9964e-23a3-438a-a726-8d60ebbb1c3c",
    "securityRules": [
        {
            "access": "Allow",
            "description": null,
            "destinationAddressPrefix": "*",
            "destinationPortRange": "22",
            "direction": "Inbound",
            "etag": "W/\"9e344b60-0daa-40a6-84f9-0ebbe4a4b640\"",
            "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/securityRules/myNetworkSecurityGroupRuleSSH",
            "name": "myNetworkSecurityGroupRuleSSH",
            "priority": 1000,
            "protocol": "Tcp",
            "provisioningState": "Succeeded",
            "resourceGroup": "myResourceGroup",
            "sourceAddressPrefix": "*",
            "sourcePortRange": "*"
        },
        {
            "access": "Allow",
            "description": null,
            "destinationAddressPrefix": "*",
            "destinationPortRange": "80",
            "direction": "Inbound",
            "etag": "W/\"9e344b60-0daa-40a6-84f9-0ebbe4a4b640\"",
            "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup/securityRules/myNetworkSecurityGroupRuleWeb",
            "name": "myNetworkSecurityGroupRuleWeb",
            "priority": 1001,
            "protocol": "Tcp",
            "provisioningState": "Succeeded",
            "resourceGroup": "myResourceGroup",
            "sourceAddressPrefix": "*",
            "sourcePortRange": "*"
        }
    ],
    "subnets": null,
    "tags": null,
    "type": "Microsoft.Network/networkSecurityGroups"
}

```

## Create a virtual NIC

Virtual network interface cards (NICs) are programmatically available because you can apply rules to their use. Depending on the [VM size](#), you can attach multiple virtual NICs to a VM. In the following [az network nic create](#) command, you create a NIC named *myNic* and associate it with your network security group. The public IP address *myPublicIP* is also associated with the virtual NIC.

```

az network nic create \
    --resource-group myResourceGroup \
    --name myNic \
    --vnet-name myVnet \
    --subnet mySubnet \
    --public-ip-address myPublicIP \
    --network-security-group myNetworkSecurityGroup

```

Output:

```
{
  "NewNIC": {
    "dnsSettings": {
      "appliedDnsServers": [],
      "dnsServers": [],
      "internalDnsNameLabel": null,
      "internalDomainNameSuffix": "brqlt10lvoxedgkeuomc4pm5tb.bx.internal.cloudapp.net",
      "internalFqdn": null
    },
    "enableAcceleratedNetworking": false,
    "enableIpForwarding": false,
    "etag": "W/\"04b5ab44-d8f4-422a-9541-e5ae7de8466d\"",
    "id": ""
  },
  "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic",
  "ipConfigurations": [
    {
      "applicationGatewayBackendAddressPools": null,
      "etag": "W/\"04b5ab44-d8f4-422a-9541-e5ae7de8466d\"",
      "id": ""
    }
  ],
  "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic/ipCo
nfigurations/ipconfig1",
  "loadBalancerBackendAddressPools": null,
  "loadBalancerInboundNatRules": null,
  "name": "ipconfig1",
  "primary": true,
  "privateIpAddress": "192.168.1.4",
  "privateIpAddressVersion": "IPv4",
  "privateIpAllocationMethod": "Dynamic",
  "provisioningState": "Succeeded",
  "publicIpAddress": {
    "dnsSettings": null,
    "etag": null,
    "id": ""
  },
  "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/publicIPAddresses/myPublicIP
",
  "idleTimeoutInMinutes": null,
  "ipAddress": null,
  "ipConfiguration": null,
  "location": null,
  "name": null,
  "provisioningState": null,
  "publicIpAddressVersion": null,
  "publicIpAllocationMethod": null,
  "resourceGroup": "myResourceGroup",
  "resourceGuid": null,
  "tags": null,
  "type": null
},
"resourceGroup": "myResourceGroup",
"subnet": {
  "addressPrefix": null,
  "etag": null,
  "id": ""
},
"/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/virtualNetworks/myVnet/subne
ts/mySubnet",
  "ipConfigurations": null,
  "name": null,
  "networkSecurityGroup": null,
  "provisioningState": null,
  "resourceGroup": "myResourceGroup",
  "resourceNavigationLinks": null,
  "routeTable": null
}
],
"location": "eastus",
"macAddress": null,
"name": "myNic",
"networkSecurityGroup": {
```

```
        "defaultSecurityRules": null,
        "etag": null,
        "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNetworkSecurityGroup",
        "location": null,
        "name": null,
        "networkInterfaces": null,
        "provisioningState": null,
        "resourceGroup": "myResourceGroup",
        "resourceGuid": null,
        "securityRules": null,
        "subnets": null,
        "tags": null,
        "type": null
    },
    "primary": null,
    "provisioningState": "Succeeded",
    "resourceGroup": "myResourceGroup",
    "resourceGuid": "b3dbaa0e-2cf2-43be-a814-5cc49fea3304",
    "tags": null,
    "type": "Microsoft.Network/networkInterfaces",
    "virtualMachine": null
}
}
```

## Create an availability set

Availability sets help spread your VMs across fault domains and update domains. Even though you only create one VM right now, it's best practice to use availability sets to make it easier to expand in the future.

Fault domains define a grouping of virtual machines that share a common power source and network switch. By default, the virtual machines that are configured within your availability set are separated across up to three fault domains. A hardware issue in one of these fault domains does not affect every VM that is running your app.

Update domains indicate groups of virtual machines and underlying physical hardware that can be rebooted at the same time. During planned maintenance, the order in which update domains are rebooted might not be sequential, but only one update domain is rebooted at a time.

Azure automatically distributes VMs across the fault and update domains when placing them in an availability set. For more information, see [managing the availability of VMs](#).

Create an availability set for your VM with [az vm availability-set create](#). The following example creates an availability set named *myAvailabilitySet*.

```
az vm availability-set create \
--resource-group myResourceGroup \
--name myAvailabilitySet
```

The output notes fault domains and update domains:

```
{  
  "id":  
    "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Compute/availabilitySets/myAvailabilitySet",  
  "location": "eastus",  
  "managed": null,  
  "name": "myAvailabilitySet",  
  "platformFaultDomainCount": 2,  
  "platformUpdateDomainCount": 5,  
  "resourceGroup": "myResourceGroup",  
  "sku": {  
    "capacity": null,  
    "managed": true,  
    "tier": null  
  },  
  "statuses": null,  
  "tags": {},  
  "type": "Microsoft.Compute/availabilitySets",  
  "virtualMachines": []  
}
```

## Create a VM

You've created the network resources to support Internet-accessible VMs. Now create a VM and secure it with an SSH key. In this example, let's create an Ubuntu VM based on the most recent LTS. You can find additional images with [az vm image list](#), as described in [finding Azure VM images](#).

Specify an SSH key to use for authentication. If you do not have an SSH public key pair, you can [create them](#) or use the `--generate-ssh-keys` parameter to create them for you. If you already have a key pair, this parameter uses existing keys in `~/.ssh`.

Create the VM by bringing all the resources and information together with the [az vm create](#) command. The following example creates a VM named *myVM*.

```
az vm create \  
  --resource-group myResourceGroup \  
  --name myVM \  
  --location eastus \  
  --availability-set myAvailabilitySet \  

```

SSH to your VM with the DNS entry you provided when you created the public IP address. This `fqdn` is shown in the output as you create your VM:

```
{  
  "fqdns": "mypublicdns.eastus.cloudapp.azure.com",  
  "id":  
    "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",  
  "location": "eastus",  
  "macAddress": "00-0D-3A-13-71-C8",  
  "powerState": "VM running",  
  "privateIpAddress": "192.168.1.5",  
  "publicIpAddress": "13.90.94.252",  
  "resourceGroup": "myResourceGroup"  
}
```

```
ssh azureuser@mypublicdns.eastus.cloudapp.azure.com
```

Output:

```
The authenticity of host 'mypublicdns.eastus.cloudapp.azure.com (13.90.94.252)' can't be established.  
ECDSA key fingerprint is SHA256:SyLlNP80Um6XRTvWiFaNz+H+1jcrKB1IiNgCDDJRj6A.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added 'mypublicdns.eastus.cloudapp.azure.com,13.90.94.252' (ECDSA) to the list of known  
hosts.  
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.11.0-1016-azure x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/advantage  
  
Get cloud support with Ubuntu Advantage Cloud Guest:  
https://www.ubuntu.com/business/services/cloud  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
azureuser@myVM:~$
```

You can install NGINX and see the traffic flow to the VM. Install NGINX as follows:

```
sudo apt-get install -y nginx
```

To see the default NGINX site in action, open your web browser and enter your FQDN:



## Export as a template

What if you now want to create an additional development environment with the same parameters, or a production environment that matches it? Resource Manager uses JSON templates that define all the parameters for your environment. You build out entire environments by referencing this JSON template. You can [build JSON](#)

[templates manually](#) or export an existing environment to create the JSON template for you. Use [az group export](#) to export your resource group as follows:

```
az group export --name myResourceGroup > myResourceGroup.json
```

This command creates the `myResourceGroup.json` file in your current working directory. When you create an environment from this template, you are prompted for all the resource names. You can populate these names in your template file by adding the `--include-parameter-default-value` parameter to the `az group export` command. Edit your JSON template to specify the resource names, or [create a parameters.json file](#) that specifies the resource names.

To create an environment from your template, use [az deployment group create](#) as follows:

```
az deployment group create \
--resource-group myNewResourceGroup \
--template-file myResourceGroup.json
```

You might want to read [more about how to deploy from templates](#). Learn about how to incrementally update environments, use the parameters file, and access templates from a single storage location.

## Next steps

Now you're ready to begin working with multiple networking components and VMs. You can use this sample environment to build out your application by using the core components introduced here.

# How to create a Linux virtual machine with Azure Resource Manager templates

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Learn how to create a Linux virtual machine (VM) by using an Azure Resource Manager template and the Azure CLI from the Azure Cloud shell. To create a Windows virtual machine, see [Create a Windows virtual machine from a Resource Manager template](#).

An alternative is to deploy the template from the Azure portal. To open the template in the portal, select the **Deploy to Azure** button.



## Templates overview

Azure Resource Manager templates are JSON files that define the infrastructure and configuration of your Azure solution. By using a template, you can repeatedly deploy your solution throughout its lifecycle and have confidence your resources are deployed in a consistent state. To learn more about the format of the template and how you construct it, see [Quickstart: Create and deploy Azure Resource Manager templates by using the Azure portal](#). To view the JSON syntax for resources types, see [Define resources in Azure Resource Manager templates](#).

## Create a virtual machine

Creating an Azure virtual machine usually includes two steps:

1. Create a resource group. An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine.
2. Create a virtual machine.

The following example creates a VM from an [Azure Quickstart template](#). This template creates an Azure Generation 2 VM by default. See [Support for generation 2 VMs on Azure](#) to learn more about Azure Generation 2 VMs. Only SSH authentication is allowed for this deployment. When prompted, provide the value of your own SSH public key, such as the contents of `~/.ssh/id_rsa.pub`. If you need to create an SSH key pair, see [How to create and use an SSH key pair for Linux VMs in Azure](#). Here is a copy of the template:

```
{  
    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
    "contentVersion": "1.0.0.0",  
    "parameters": {  
        " projectName": {  
            "type": "string",  
            "metadata": {  
                "description": "Specifies a name for generating resource names."  
            }  
        },  
        " location": {  
            "type": "string",  
            "defaultValue": "[resourceGroup().location]",  
            "metadata": {  
                "description": "Specifies the location for all resources."  
            }  
        }  
    }  
}
```

```

        },
        "adminUsername": {
            "type": "string",
            "metadata": {
                "description": "Specifies a username for the Virtual Machine."
            }
        },
        "adminPublicKey": {
            "type": "string",
            "metadata": {
                "description": "Specifies the SSH rsa public key file as a string. Use \"ssh-keygen -t rsa -b 2048\" to generate your SSH key pairs."
            }
        },
        "vmSize": {
            "type": "string",
            "defaultValue": "Standard_D2s_v3",
            "metadata": {
                "description": "description"
            }
        }
    },
    "variables": {
        "vNetName": "[concat(parameters('projectName'), '-vnet')]",
        "vNetAddressPrefixes": "10.0.0.0/16",
        "vNetSubnetName": "default",
        "vNetSubnetAddressPrefix": "10.0.0.0/24",
        "vmName": "[concat(parameters('projectName'), '-vm')]",
        "publicIPAddressName": "[concat(parameters('projectName'), '-ip')]",
        "networkInterfaceName": "[concat(parameters('projectName'), '-nic')]",
        "networkSecurityGroupName": "[concat(parameters('projectName'), '-nsg')]",
        "networkSecurityGroupName2": "[concat(variables('vNetSubnetName'), '-nsg')]"
    },
    "resources": [
        {
            "type": "Microsoft.Network/networkSecurityGroups",
            "apiVersion": "2020-05-01",
            "name": "[variables('networkSecurityGroupName')]",
            "location": "[parameters('location')]",
            "properties": {
                "securityRules": [
                    {
                        "name": "ssh_rule",
                        "properties": {
                            "description": "Locks inbound down to ssh default port 22.",
                            "protocol": "Tcp",
                            "sourcePortRange": "*",
                            "destinationPortRange": "22",
                            "sourceAddressPrefix": "*",
                            "destinationAddressPrefix": "*",
                            "access": "Allow",
                            "priority": 123,
                            "direction": "Inbound"
                        }
                    }
                ]
            }
        },
        {
            "type": "Microsoft.Network/publicIPAddresses",
            "apiVersion": "2020-05-01",
            "name": "[variables('publicIPAddressName')]",
            "location": "[parameters('location')]",
            "properties": {
                "publicIPAllocationMethod": "Dynamic"
            },
            "sku": {
                "name": "Basic"
            }
        }
    ]
}

```

```

    }
},
{
  "comments": "Simple Network Security Group for subnet [variables('vNetSubnetName')]",
  "type": "Microsoft.Network/networkSecurityGroups",
  "apiVersion": "2020-05-01",
  "name": "[variables('networkSecurityGroupName2')]",
  "location": "[parameters('location')]",
  "properties": {
    "securityRules": [
      {
        "name": "default-allow-22",
        "properties": {
          "priority": 1000,
          "access": "Allow",
          "direction": "Inbound",
          "destinationPortRange": "22",
          "protocol": "Tcp",
          "sourceAddressPrefix": "*",
          "sourcePortRange": "*",
          "destinationAddressPrefix": "*"
        }
      }
    ]
  }
},
{
  "type": "Microsoft.Network/virtualNetworks",
  "apiVersion": "2020-05-01",
  "name": "[variables('vNetName')]",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName2'))]"
  ],
  "properties": {
    "addressSpace": {
      "addressPrefixes": [
        "[variables('vNetAddressPrefixes')]"
      ]
    },
    "subnets": [
      {
        "name": "[variables('vNetSubnetName')]",
        "properties": {
          "addressPrefix": "[variables('vNetSubnetAddressPrefix')]",
          "networkSecurityGroup": {
            "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('networkSecurityGroupName2'))]"
          }
        }
      }
    ]
  }
},
{
  "type": "Microsoft.Network/networkInterfaces",
  "apiVersion": "2020-05-01",
  "name": "[variables('networkInterfaceName')]",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[resourceId('Microsoft.Network/publicIPAddresses', variables('publicIPAddressName'))]",
    "[resourceId('Microsoft.Network/virtualNetworks', variables('vNetName'))]",
    "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"
  ],
  "properties": {
    "ipConfigurations": [
      {
        "name": "ipconfig1",
        "properties": {

```

```

        "privateIPAllocationMethod": "Dynamic",
        "publicIPAddress": {
            "id": "[resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))]"
        },
        "subnet": {
            "id": "[resourceId('Microsoft.Network/virtualNetworks/subnets', variables('vNetName'),
variables('vNetSubnetName'))]"
        }
    }
},
{
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2021-11-01",
    "name": "[variables('vmName')]",
    "location": "[parameters('location')]",
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]"
    ],
    "properties": {
        "hardwareProfile": {
            "vmSize": "[parameters('vmSize')]"
        },
        "osProfile": {
            "computerName": "[variables('vmName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "linuxConfiguration": {
                "disablePasswordAuthentication": true,
                "ssh": {
                    "publicKeys": [
                        {
                            "path": "[concat('/home/', parameters('adminUsername'), '/.ssh/authorized_keys')]",
                            "keyData": "[parameters('adminPublicKey')]"
                        }
                    ]
                }
            }
        },
        "storageProfile": {
            "imageReference": {
                "publisher": "Canonical",
                "offer": "UbuntuServer",
                "sku": "18_04-lts-gen2",
                "version": "latest"
            },
            "osDisk": {
                "createOption": "fromImage"
            }
        },
        "networkProfile": {
            "networkInterfaces": [
                {
                    "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]"
                }
            ]
        }
    }
}
]
}

```

To run the CLI script, Select **Try it** to open the Azure Cloud shell. To paste the script, right-click the shell, and then select **Paste**:

```

echo "Enter the Resource Group name:" &&
read resourceGroupName &&
echo "Enter the location (i.e. centralus):" &&
read location &&
echo "Enter the project name (used for generating resource names):" &&
read projectName &&
echo "Enter the administrator username:" &&
read username &&
echo "Enter the SSH public key:" &&
read key &&
az group create --name $resourceGroupName --location "$location" &&
az deployment group create --resource-group $resourceGroupName --template-uri
https://raw.githubusercontent.com/azure/azure-quickstart-templates/master/quickstarts/microsoft.compute/vm-
sshkey/azuredeploy.json --parameters projectName=$projectName adminUsername=$username adminPublicKey="$key"
&&
az vm show --resource-group $resourceGroupName --name "$projectName-vm" --show-details --query publicIps --
output tsv

```

The last Azure CLI command shows the public IP address of the newly created VM. You need the public IP address to connect to the virtual machine. See the next section of this article.

In the previous example, you specified a template stored in GitHub. You can also download or create a template and specify the local path with the `--template-file` parameter.

Here are some additional resources:

- To learn how to develop Resource Manager templates, see [Azure Resource Manager documentation](#).
- To see the Azure virtual machine schemas, see [Azure template reference](#).
- To see more virtual machine template samples, see [Azure Quickstart templates](#).

## Connect to virtual machine

You can then SSH to your VM as normal. Provide your own public IP address from the preceding command:

```
ssh <adminUsername>@<ipAddress>
```

## Next steps

In this example, you created a basic Linux VM. For more Resource Manager templates that include application frameworks or create more complex environments, browse the [Azure Quickstart templates](#).

To learn more about creating templates, view the JSON syntax and properties for the resources types you deployed:

- [Microsoft.Network/networkSecurityGroups](#)
- [Microsoft.Network/publicIPAddresses](#)
- [Microsoft.Network/virtualNetworks](#)
- [Microsoft.Network/networkInterfaces](#)
- [Microsoft.Compute/virtualMachines](#)

# Create a Linux virtual machine that uses SSH authentication with the REST API

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

A Linux virtual machine (VM) in Azure consists of various resources such as disks and network interfaces and defines parameters such as location, size and operating system image and authentication settings.

You can create a Linux VM via the Azure portal, Azure CLI 2.0, many Azure SDKs, Azure Resource Manager templates and many third-party tools such as Ansible or Terraform. All these tools ultimately use the REST API to create the Linux VM.

This article shows you how to use the REST API to create a Linux VM running Ubuntu 18.04-LTS with managed disks and SSH authentication.

## Before you start

Before you create and submit the request, you will need:

- The `{subscription-id}` for your subscription
  - If you have multiple subscriptions, see [Working with multiple subscriptions](#)
- A `{resourceGroupName}` you've created ahead of time
- A [virtual network interface](#) in the same resource group
- An SSH key pair (you can [generate a new one](#) if you don't have one)

## Request basics

To create or update a virtual machine, use the following *PUT* operation:

```
PUT https://management.azure.com/subscriptions/{subscription-id}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmName}?api-version=2017-12-01
```

In addition to the `{subscription-id}` and `{resourceGroupName}` parameters, you'll need to specify the `{vmName}` (`api-version` is optional, but this article was tested with `api-version=2017-12-01`)

The following headers are required:

REQUEST HEADER	DESCRIPTION
<code>Content-Type:</code>	Required. Set to <code>application/json</code> .
<code>Authorization:</code>	Required. Set to a valid <code>Bearer</code> <a href="#">access token</a> .

For general information about working with REST API requests, see [Components of a REST API request/response](#).

## Create the request body

The following common definitions are used to build a request body:

NAME	REQUIRED	TYPE	DESCRIPTION
location	True	string	Resource location.
name		string	Name for the virtual machine.
properties.hardwareProfile		HardwareProfile	Specifies the hardware settings for the virtual machine.
properties.storageProfile		StorageProfile	Specifies the storage settings for the virtual machine disks.
properties.osProfile		OSProfile	Specifies the operating system settings for the virtual machine.
properties.networkProfile		NetworkProfile	Specifies the network interfaces of the virtual machine.

An example request body is below. Make sure you specify the VM name in the `{computerName}` and `{name}` parameters, the name of the network interface you've created under `networkInterfaces`, your username in `adminUsername` and `path`, and the `public` portion of your SSH keypair (located in, for example, `~/.ssh/id_rsa.pub`) in `keyData`. Other parameters you might want to modify include `location` and `vmSize`.

```
{
  "location": "eastus",
  "name": "{vmName}",
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_DS1_v2"
    },
    "storageProfile": {
      "imageReference": {
        "sku": "18.04-LTS",
        "publisher": "Canonical",
        "version": "latest",
        "offer": "UbuntuServer"
      },
      "osDisk": {
        "caching": "ReadWrite",
        "managedDisk": {
          "storageAccountType": "Premium_LRS"
        },
        "name": "myVMosdisk",
        "createOption": "FromImage"
      }
    },
    "osProfile": {
      "adminUsername": "{your-username}",
      "computerName": "{vmName}",
      "linuxConfiguration": {
        "ssh": {
          "publicKeys": [
            {
              "path": "/home/{your-username}/.ssh/authorized_keys",
              "keyData": "ssh-rsa AAAAB3NzaC1{snip}mf69/J1"
            }
          ]
        },
        "disablePasswordAuthentication": true
      }
    },
    "networkProfile": {
      "networkInterfaces": [
        {
          "id": "/subscriptions/{subscription-
id}/resourceGroups/{resourceGroupName}/providers/Microsoft.Network/networkInterfaces/{existing-nic-name}",
          "properties": {
            "primary": true
          }
        }
      ]
    }
  }
}
```

For a complete list of the available definitions in the request body, see [Virtual machines create or update request body definitions](#).

## Sending the request

You may use the client of your preference for sending this HTTP request. You may also use an [in-browser tool](#) by clicking the **Try it** button.

### Responses

There are two successful responses for the operation to create or update a virtual machine:

NAME	TYPE	DESCRIPTION
200 OK	<a href="#">VirtualMachine</a>	OK
201 Created	<a href="#">VirtualMachine</a>	Created

A condensed `201 Created` response from the previous example request body that creates a VM shows a `vmId` has been assigned and the `provisioningState` is `Creating`.

```
{
  "vmId": "e0de9b84-a506-4b95-9623-00a425d05c90",
  "provisioningState": "Creating"
}
```

For more information about REST API responses, see [Process the response message](#).

## Next steps

For more information on the Azure REST APIs or other management tools such as Azure CLI or Azure PowerShell, see the following:

- [Azure Compute provider REST API](#)
- [Get started with Azure REST API](#)
- [Azure CLI](#)
- [Azure PowerShell module](#)

# Tutorial: Install a LAMP stack on an Azure Linux VM

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This article walks you through how to deploy an Apache web server, MySQL, and PHP (the LAMP stack) on an Ubuntu VM in Azure. To see the LAMP server in action, you can optionally install and configure a WordPress site. In this tutorial you learn how to:

- Create an Ubuntu VM
- Open port 80 for web traffic
- Install Apache, MySQL, and PHP
- Verify installation and configuration
- Install WordPress

This setup is for quick tests or proof of concept. For more on the LAMP stack, including recommendations for a production environment, see the [Ubuntu documentation](#).

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create a resource group

Create a resource group with the [az group create](#) command. An Azure resource group is a logical container into which Azure resources are deployed and managed.

The following example creates a resource group named *myResourceGroup* in the *eastus* location.

```
az group create --name myResourceGroup --location eastus
```

## Create a virtual machine

Create a VM with the [az vm create](#) command.

The following example creates a VM named *myVM* and creates SSH keys if they do not already exist in a default key location. To use a specific set of keys, use the `--ssh-key-value` option. The command also sets *azureuser* as an administrator user name. You use this name later to connect to the VM.

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image UbuntuLTS \
--admin-username azureuser \
--generate-ssh-keys
```

When the VM has been created, the Azure CLI shows information similar to the following example. Take note of the `publicIpAddress`. This address is used to access the VM in later steps.

```
{  
  "fqdns": "",  
  "id": "/subscriptions/<subscription  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",  
  "location": "eastus",  
  "macAddress": "00-0D-3A-23-9A-49",  
  "powerState": "VM running",  
  "privateIpAddress": "10.0.0.4",  
  "publicIpAddress": "40.68.254.142",  
  "resourceGroup": "myResourceGroup"  
}
```

## Open port 80 for web traffic

By default, only SSH connections are allowed into Linux VMs deployed in Azure. Because this VM is going to be a web server, you need to open port 80 from the internet. Use the [az vm open-port](#) command to open the desired port.

```
az vm open-port --port 80 --resource-group myResourceGroup --name myVM
```

For more information about opening ports to your VM, see [Open ports](#).

## SSH into your VM

If you don't already know the public IP address of your VM, run the [az network public-ip list](#) command. You need this IP address for several later steps.

```
az network public-ip list --resource-group myResourceGroup --query []. ipAddress
```

Use the following command to create an SSH session with the virtual machine. Substitute the correct public IP address of your virtual machine. In this example, the IP address is *40.68.254.142*. *azureuser* is the administrator user name set when you created the VM.

```
ssh azureuser@40.68.254.142
```

## Install Apache, MySQL, and PHP

Run the following command to update Ubuntu package sources and install Apache, MySQL, and PHP. Note the caret (^) at the end of the command, which is part of the `lamp-server^` package name.

```
sudo apt update && sudo apt install lamp-server^
```

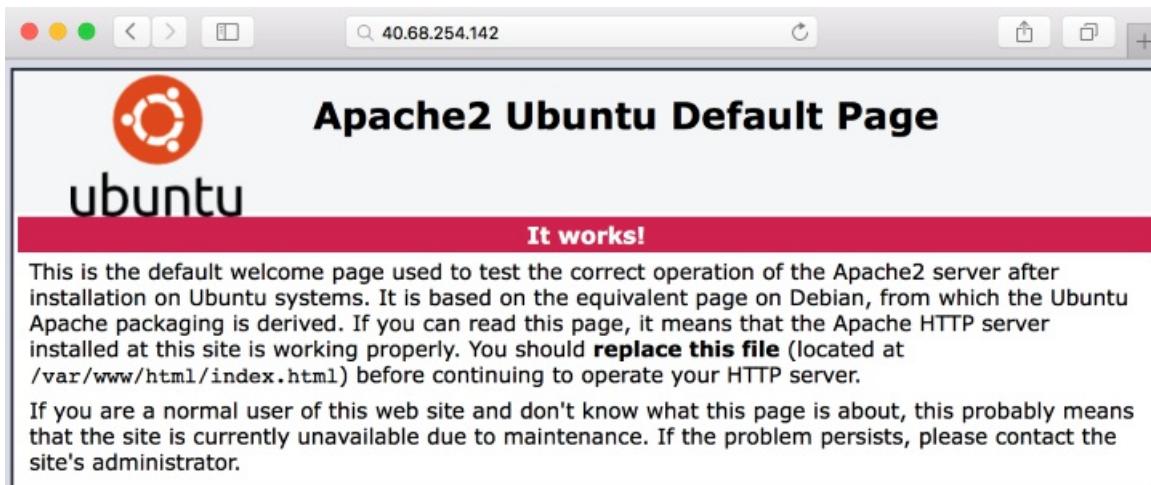
You are prompted to install the packages and other dependencies. This process installs the minimum required PHP extensions needed to use PHP with MySQL.

## Verify Apache

Check the version of Apache with the following command:

```
apache2 -v
```

With Apache installed, and port 80 open to your VM, the web server can now be accessed from the internet. To view the Apache2 Ubuntu Default Page, open a web browser, and enter the public IP address of the VM. Use the public IP address you used to SSH to the VM:



## Verify and secure MySQL

Check the version of MySQL with the following command (note the capital `v` parameter):

```
mysql -v
```

To help secure the installation of MySQL, including setting a root password, run the `mysql_secure_installation` script.

```
sudo mysql_secure_installation
```

You can optionally set up the Validate Password Plugin (recommended). Then, set a password for the MySQL root user, and configure the remaining security settings for your environment. We recommend that you answer "Y" (yes) to all questions.

If you want to try MySQL features (create a MySQL database, add users, or change configuration settings), login to MySQL. This step is not required to complete this tutorial.

```
sudo mysql -u root -p
```

When done, exit the mysql prompt by typing `\q`.

## Verify PHP

Check the version of PHP with the following command:

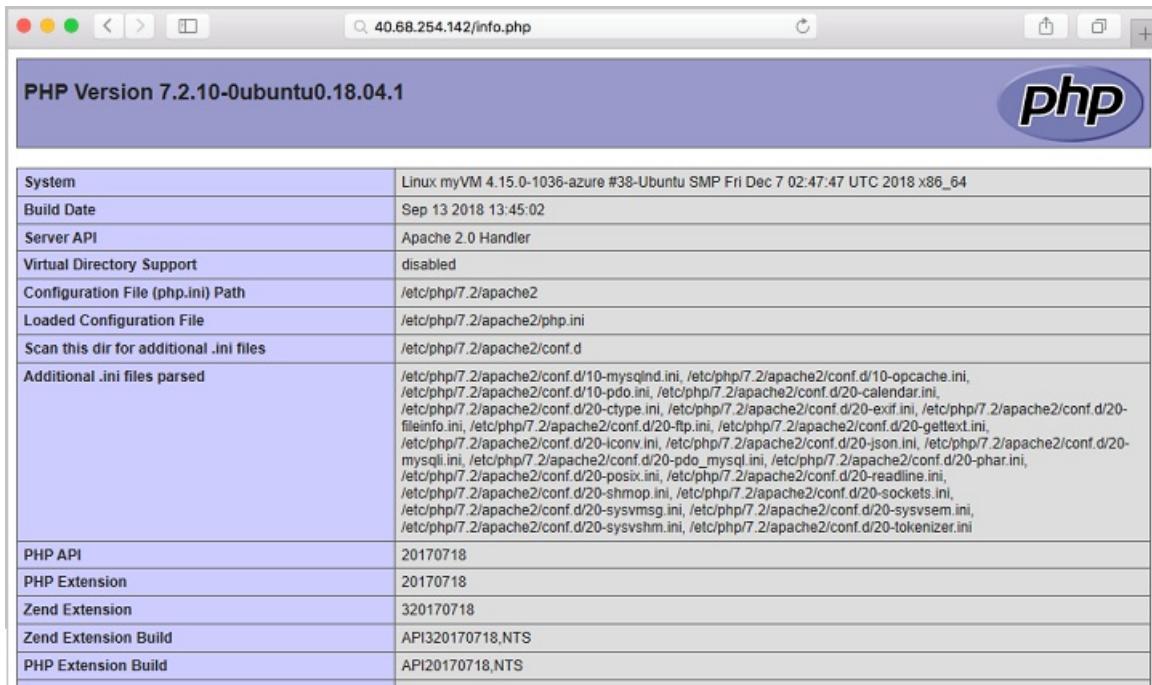
```
php -v
```

If you want to test further, create a quick PHP info page to view in a browser. The following command creates the PHP info page:

```
sudo sh -c 'echo "<?php phpinfo(); ?>" > /var/www/html/info.php'
```

Now you can check the PHP info page you created. Open a browser and go to

`http://yourPublicIPAddress/info.php`. Substitute the public IP address of your VM. It should look similar to this image.



A screenshot of a web browser window displaying PHP version information. The title bar shows the URL `40.68.254.142/info.php`. The page header includes the text "PHP Version 7.2.10-0ubuntu0.18.04.1" and the PHP logo. Below this is a table containing various PHP configuration details:

System	Linux myVM 4.15.0-1036-azure #38-Ubuntu SMP Fri Dec 7 02:47:47 UTC 2018 x86_64
Build Date	Sep 13 2018 13:45:02
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.2/apache2
Loaded Configuration File	/etc/php/7.2/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.2/apache2/conf.d
Additional .ini files parsed	/etc/php/7.2/apache2/conf.d/10-mysqlind.ini, /etc/php/7.2/apache2/conf.d/10-opcache.ini, /etc/php/7.2/apache2/conf.d/10-pdo.ini, /etc/php/7.2/apache2/conf.d/20-calendar.ini, /etc/php/7.2/apache2/conf.d/20-ctype.ini, /etc/php/7.2/apache2/conf.d/20-exif.ini, /etc/php/7.2/apache2/conf.d/20-fileinfo.ini, /etc/php/7.2/apache2/conf.d/20-ftp.ini, /etc/php/7.2/apache2/conf.d/20-gettext.ini, /etc/php/7.2/apache2/conf.d/20-iconv.ini, /etc/php/7.2/apache2/conf.d/20-json.ini, /etc/php/7.2/apache2/conf.d/20-mysqli.ini, /etc/php/7.2/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.2/apache2/conf.d/20-phar.ini, /etc/php/7.2/apache2/conf.d/20-posix.ini, /etc/php/7.2/apache2/conf.d/20-readline.ini, /etc/php/7.2/apache2/conf.d/20-shmop.ini, /etc/php/7.2/apache2/conf.d/20-sockets.ini, /etc/php/7.2/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.2/apache2/conf.d/20-sysvsem.ini, /etc/php/7.2/apache2/conf.d/20-sysvshm.ini, /etc/php/7.2/apache2/conf.d/20-tokenizer.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718,NTS
PHP Extension Build	API20170718,NTS

## Install WordPress

If you want to try your stack, install a sample app. As an example, the following steps install the open source [WordPress](#) platform to create websites and blogs. Other workloads to try include [Drupal](#) and [Moodle](#).

This WordPress setup is only for proof of concept. To install the latest WordPress in production with recommended security settings, see the [WordPress documentation](#).

### Install the WordPress package

Run the following command:

```
sudo apt install wordpress
```

### Configure WordPress

Configure WordPress to use MySQL and PHP.

In a working directory, create a text file `wordpress.sql` to configure the MySQL database for WordPress:

```
sudo sensible-editor wordpress.sql
```

Add the following commands, substituting a database password of your choice for `yourPassword` (leave other values unchanged). If you previously set up a MySQL security policy to validate password strength, make sure the password meets the strength requirements. Save the file.

```
CREATE DATABASE wordpress;
GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,ALTER
ON wordpress.*
TO wordpress@localhost
IDENTIFIED BY 'yourPassword';
```

Run the following command to create the database:

```
cat wordpress.sql | sudo mysql --defaults-extra-file=/etc/mysql/debian.cnf
```

Because the file `wordpress.sql` contains database credentials, delete it after use:

```
sudo rm wordpress.sql
```

To configure PHP, run the following command to open a text editor of your choice and create the file

```
/etc/wordpress/config-localhost.php :
```

```
sudo sensible-editor /etc/wordpress/config-localhost.php
```

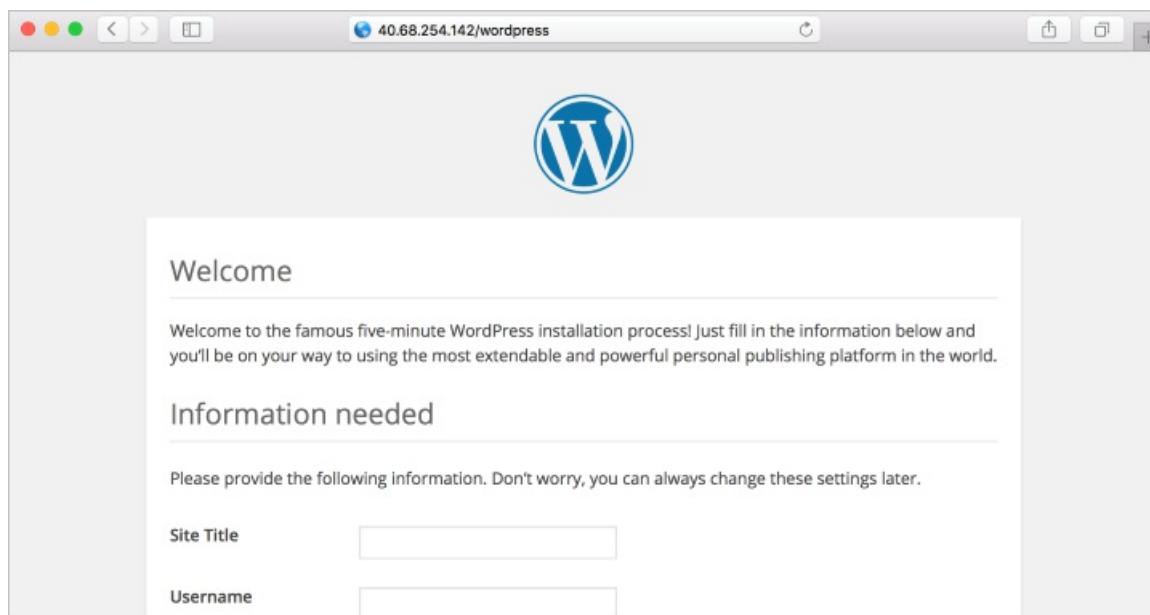
Copy the following lines to the file, substituting your WordPress database password for `yourPassword` (leave other values unchanged). Then save the file.

```
<?php
define('DB_NAME', 'wordpress');
define('DB_USER', 'wordpress');
define('DB_PASSWORD', 'yourPassword');
define('DB_HOST', 'localhost');
define('WP_CONTENT_DIR', '/usr/share/wordpress/wp-content');
?>
```

Move the WordPress installation to the web server document root:

```
sudo ln -s /usr/share/wordpress /var/www/html/wordpress
sudo mv /etc/wordpress/config-localhost.php /etc/wordpress/config-default.php
```

Now you can complete the WordPress setup and publish on the platform. Open a browser and go to `http://yourPublicIPAddress/wordpress`. Substitute the public IP address of your VM. It should look similar to this image.



## Next steps

In this tutorial, you deployed a LAMP server in Azure. You learned how to:

- Create an Ubuntu VM
- Open port 80 for web traffic
- Install Apache, MySQL, and PHP
- Verify installation and configuration
- Install WordPress on the LAMP server

Advance to the next tutorial to learn how to secure web servers with TLS/SSL certificates.

[Secure web server with TLS](#)

# Configure the rolling deployment strategy for Azure Linux virtual machines

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Azure Pipelines provides a fully featured set of CI/CD automation tools for deployments to virtual machines. This article will show you how to set up a classic release pipeline that uses the rolling strategy to deploy your web applications to Linux virtual machines.

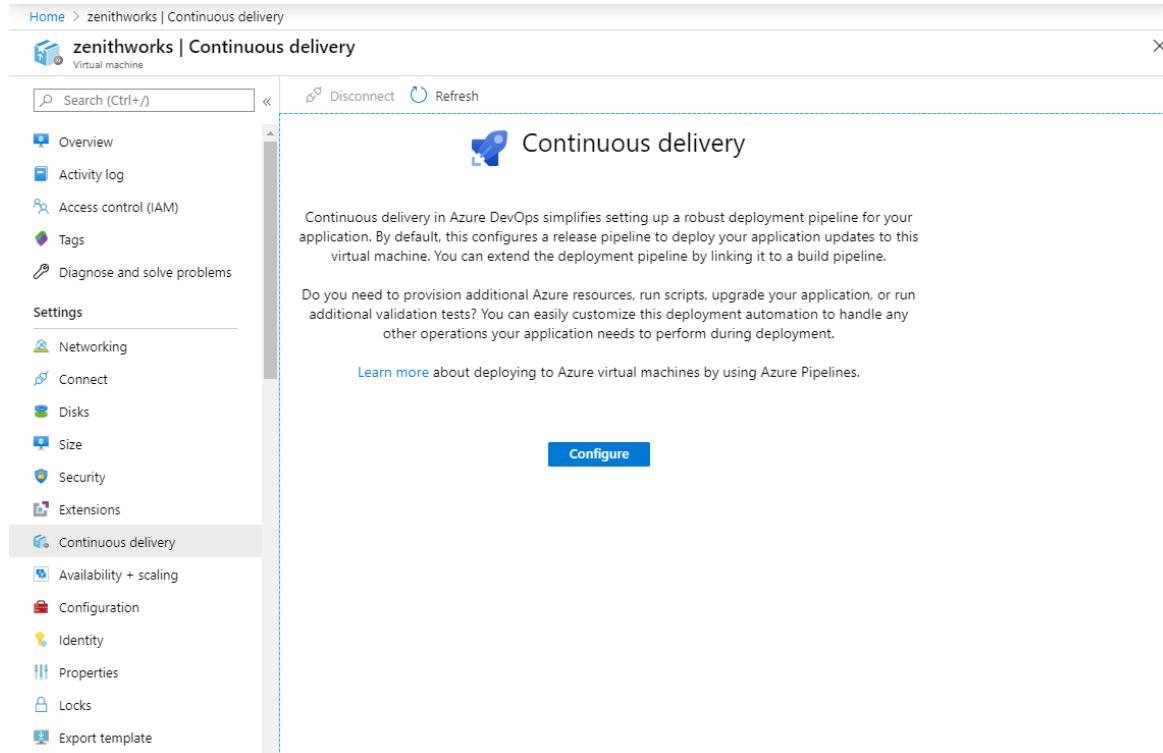
## Rolling deployments

In each iteration, a rolling deployment replaces instances of an application's previous version. It replaces them with instances of the new version on a fixed set of machines (rolling set). The following walk-through shows how to configure a rolling update to virtual machines.

Using **Continuous-delivery**, you can configure rolling updates to your virtual machines within the Azure portal.

[!IMPORTANT] Virtual Machine's Continuous delivery setting will be retired on March 31, 2023. [Learn more](#)

1. Sign in to [Azure portal](#) and navigate to a virtual machine.
2. Select **Continuous delivery**, and then select **Configure**.



3. Select your **Azure DevOps Organization** and your **Project** from the dropdown menu or **Create** a new one.
4. Select your **Deployment group** from the dropdown menu or **Create** a new one.
5. Select your **Build pipeline**.

6. Select **Deployment strategy**, and then select **Rolling**.

The screenshot shows the Azure portal interface for configuring a continuous delivery pipeline. On the left, a sidebar lists various settings like Overview, Activity log, and Tags. The 'Continuous delivery' section is selected. The main pane displays the 'Continuous delivery' configuration dialog. It includes fields for 'Azure DevOps Organization' (set to 'Use existing' and 'amja'), 'Project' (set to 'zenithworks'), 'Deployment group' (set to 'Use existing'), 'Deployment group name' (set to 'production-dg'), 'Build pipeline' (set to 'zw-core-Cl'), and 'Deployment strategy' (set to 'Rolling'). A dropdown menu also lists 'Canary' and 'Blue-Green' strategies. At the bottom right of the dialog is an 'OK' button.

7. Optionally, you can tag each machine with its role such as *web* or *db*. These tags help you target only VMs that have a specific role.

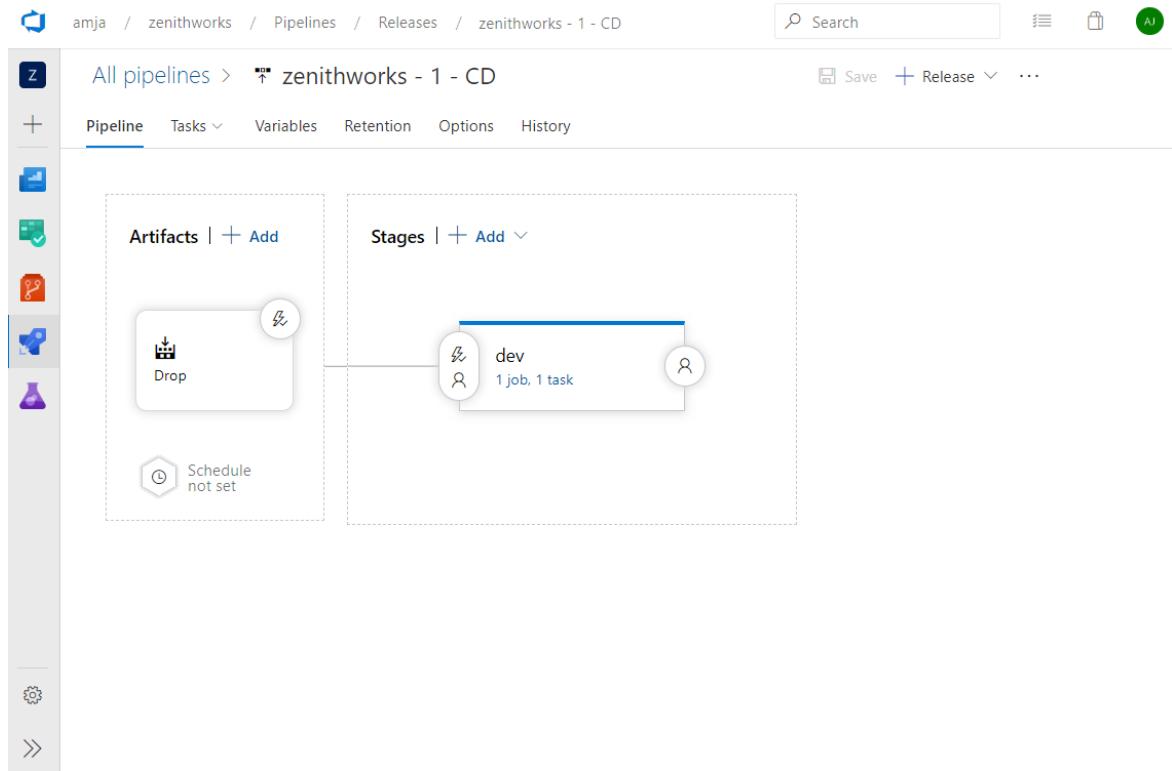
8. Select **OK** to configure the continuous delivery pipeline.

9. After completion, your continuous delivery pipeline should look similar to the following.

The screenshot shows the Azure portal interface after the continuous delivery pipeline has been configured. The sidebar and main pane are identical to the previous screenshot, but the main pane now displays the 'Deployment history' section. It shows a single deployment entry from 4/3/2020: 'zenithworks - 1 - CD > Release-1 / dev' was deployed at 7:29:43 PM on 4/3/2020, took 00:00:03, and was successful (indicated by a green checkmark). The deployment was triggered from branch 'master'.

10. If you want to configure multiple VMs, repeat steps 2 through 4 for the other VMs. If you use the same deployment group that already has a configured pipeline, the new VMs will just be added to the deployment group and no new pipelines will be created.

11. Select the link to navigate to your pipeline, and then select **Edit** to modify the pipeline definition.



12. Select the tasks in the **dev** stage to navigate to the pipeline tasks, and then select **Deploy**.

The screenshot shows the Azure Pipelines interface with the 'Tasks' tab selected. On the left, there's a sidebar with icons for Artifacts, Stages, Variables, Retention, Options, and History. The main content area shows a list of tasks under the 'dev' stage. One task, 'Deploy', is highlighted. To the right, the 'Deploy' task's configuration is shown. It has a 'Display name' of 'Deploy', a 'Deployment targets' section with a 'Deployment group' of 'production-dg', and a 'Targets to deploy to in parallel' section where 'One target at a time' is selected. There's also a 'Timeout' field set to '0'.

13. You can specify the number of target machines to deploy to in parallel in each iteration. If you want to deploy to multiple machines, you can specify the number of machines as a percentage by using the slider.
14. The **Execute Deploy Script** task will execute the deployment script located in the root folder of the published artifacts.

[← Artifacts](#)[Published](#)

Name	Size
>  buildbinaries	17 B
✓  deployscripts	17 B
□ deploy.sh	17 B

## Resources

- [Deploy to Azure virtual machines with Azure DevOps](#)
- [Deploy to Azure virtual machine scale set](#)

## Related articles

- [Configure the canary deployment strategy](#)
- [Configure the blue-green deployment strategy](#)

## Retirement

Continuous delivery setting of Virtual Machines will be retired on March 31, 2023. Please switch to directly using Azure DevOps to create customized pipelines for deployment to Azure VMs. Release pipeline [Stage Templates](#) and [Deployments Groups](#) Azure DevOps' features provide similar experiences.

### Migration Steps

There is no migration required as VM CD experience does not store any information itself, it just helps users with their Day 0 getting started experience on Azure and Azure DevOps. Users will still be able to perform all operations from Azure DevOps after retirement. You won't be able to create and view pipelines from the Azure portal anymore.

### FAQ

Where can I set up my CD pipeline after this experience is deprecated?

You won't be able to view or create Azure DevOps pipelines from an Azure portal Virtual Machine blade after retirement. You still can go to Azure DevOps portal and view or update pipelines.

Will I lose my earlier configured pipelines?

No. Your pipelines will still be available in Azure DevOps.

How can I configure different deployment strategies?

The current experience uses [deployment groups](#) to create deployment strategies. You can use deployment groups or release pipeline [Stage Templates](#) to build your pipeline with templates.

# Configure the canary deployment strategy for Azure Linux Virtual Machines

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

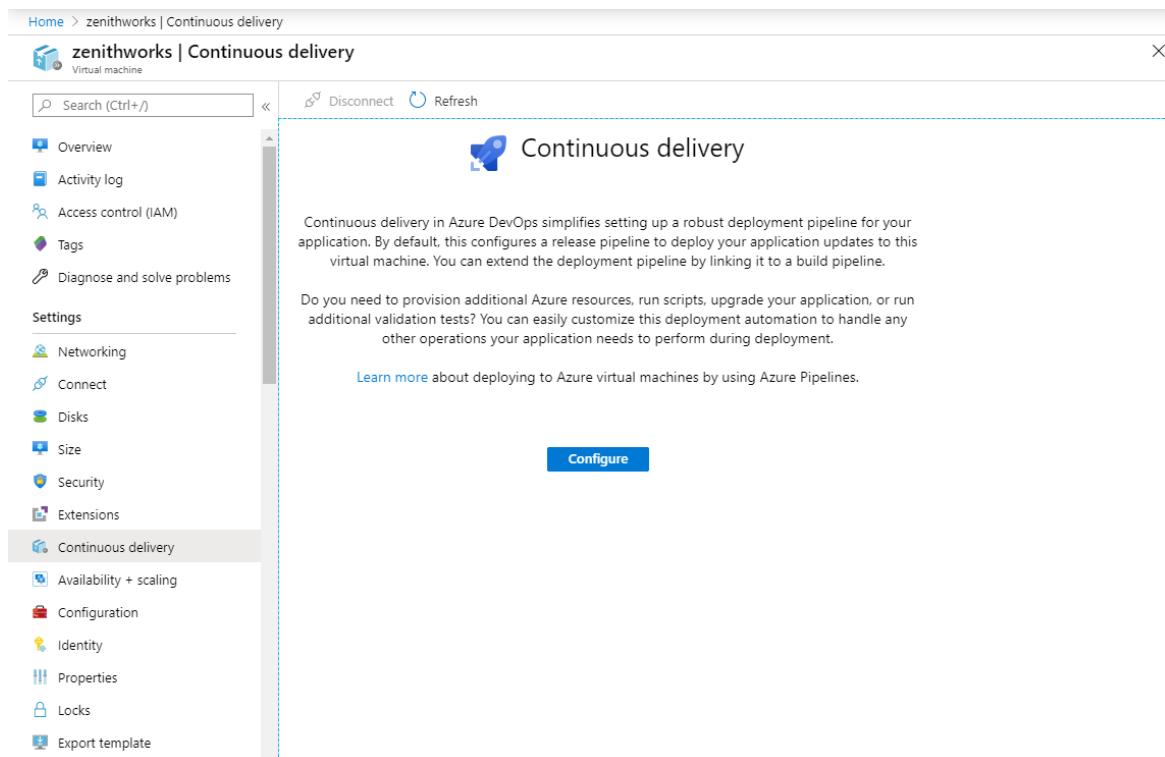
Azure Pipelines provides a fully featured set of CI/CD automation tools for deployments to virtual machines. This article will show you how to set up a classic release pipeline that uses the canary strategy to deploy web applications to Linux virtual machines.

## Canary deployments

A canary deployment reduces risk by slowly rolling out changes to a small subset of users. As you gain confidence in the new version, you can release it to more servers in your infrastructure and route more users to it.

Using the **Continuous-delivery** feature, you can use the canary strategy to deploy your application from Azure portal.

1. Sign in to [Azure portal](#) and navigate to a virtual machine.
2. Select **Continuous delivery**, and then select **Configure**.



3. In the configuration panel, select **Use existing** and select your organization/project or select **Create** and create new ones.
4. Select your **Deployment group name** from the dropdown menu or create a new one.
5. Select your **Build pipeline** from the dropdown menu.
6. Select **Deployment strategy**, and then select **Canary**.

The screenshot shows the 'Continuous delivery' configuration dialog box. In the top right corner, there is a close button (X) and a help icon (i). The title bar says 'Continuous delivery' and 'Configure a deployment pipeline'. The main area contains the following fields:

- Azure DevOps Organization \***: A dropdown menu showing 'amja'.
- Project**: A dropdown menu showing 'zenithworks'.
- Deployment group**: A dropdown menu with 'Create' and 'Use existing' options; 'Use existing' is selected.
- Deployment group name \***: A dropdown menu showing 'production-dg'.
- Build pipeline**: A dropdown menu showing 'zw-core-CI'.
- Deployment strategy \***: A dropdown menu with 'Rolling' (selected), 'Canary', and 'Blue-Green' options.

At the bottom right of the dialog box is a blue 'OK' button.

7. Add a "canary" tag to the VMs that will be used in the canary deployment.

The screenshot shows the same 'Continuous delivery' configuration dialog box as the previous one, but with a 'canary' tag added. In the bottom right corner of the dialog box, there is a 'Tags' section containing a single tag labeled 'canary'. The rest of the configuration fields are identical to the first screenshot.

8. Select OK to configure the classic release pipeline to deploy to your virtual machine.

The screenshot shows the Azure DevOps Pipelines interface. On the left, there's a sidebar with various project management and development tools like Boards, Repos, Pipelines, and Test Plans. The main area displays a release pipeline named "zenithworks - 1 - CD". The pipeline structure is shown in two columns: "Artifacts" and "Stages". The "Artifacts" column contains a single item, "Drop". The "Stages" column contains one stage named "dev", which is described as having "3 jobs, 3 tasks". Below the stage, a note says "Schedule not set".

9. Navigate to your release pipeline and then select **Edit** to view the pipeline configuration. In this example, the *dev* stage is composed of three jobs:
  - a. Deploy Canary: the application is deployed to VMs with a "canary" tag.
  - b. Wait for manual resumption: the pipeline pauses and waits for manual intervention. Before resuming the pipeline, ensure that at least one VM is tagged "prod". In the next phase, the app will be deployed only to "prod" VMs.
  - c. Deploy Prod: the application is deployed to VMs with a "prod" tag.

The screenshot shows the "Tasks" tab for the "dev" stage of the "zenithworks - 1 - CD" pipeline. The tasks listed are:

- Deploy Canary
- Execute Deploy Script
- Wait for Manual resumption
- Manual Intervention
- Deploy Prod
- Execute Deploy Script

For the "Deploy Canary" task, the configuration details on the right are:

- Deployment group \***: production-dg
- Required tags**: canary
- Targets to deploy to in parallel**:
  - Multiple
  - One target at a time
- Maximum number of targets in parallel**: 100% targets (0)
- Timeout \***: 0
- Job cancel timeout \***: 1
- Artifact download**: Specify at the time of release

## Resources

- [Deploy to Azure virtual machines with Azure DevOps](#)
- [Deploy to an Azure virtual machine scale set](#)

## Related articles

- [Configure the rolling deployment strategy](#)
- [Configure the blue-green deployment strategy](#)

# Configure the blue-green deployment strategy for Azure Linux virtual machines

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

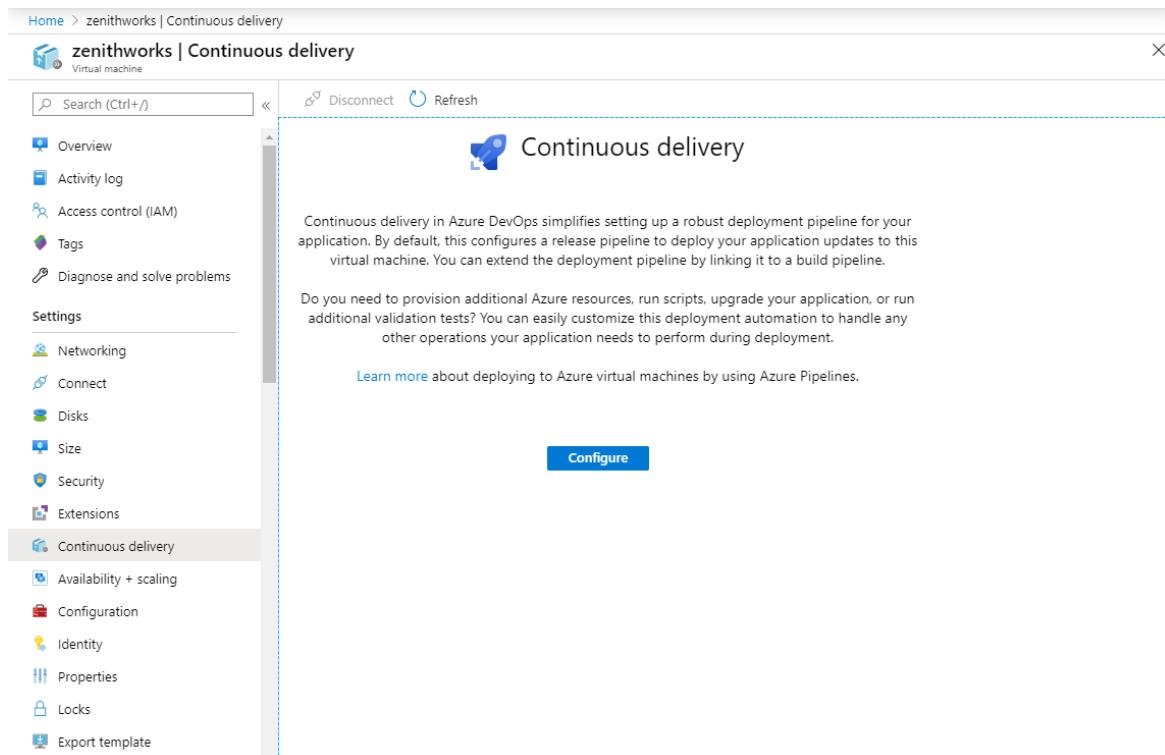
Azure Pipelines provides a fully featured set of CI/CD automation tools for deployments to virtual machines. This article will show you how to set up a classic release pipeline that uses the blue-green strategy to deploy to Linux virtual machines. Azure also supports other strategies like [rolling](#) and [canary](#) deployments.

## Blue-green deployments

A blue-green deployment is a deployment strategy where you create two separate and identical environments but only one is live at any time. This strategy is used to increase availability and reduce downtime by switching between the blue/green environments. The blue environment is usually set to run the current version of the application while the green environment is set to host the updated version. When all updates are completed, traffic is directed to the green environment and blue environment is set to idle.

Using the [Continuous-delivery](#) feature, you can use the blue-green deployment strategy to deploy to your virtual machines from Azure portal.

1. Sign in to [Azure portal](#) and navigate to a virtual machine.
2. Select [Continuous delivery](#), and then select [Configure](#).



3. In the configuration panel, select **Use existing** and select your organization/project or select **Create** and create new ones.
4. Select your **Deployment group name** from the dropdown menu or create a new one.
5. Select your **Build pipeline** from the dropdown menu.

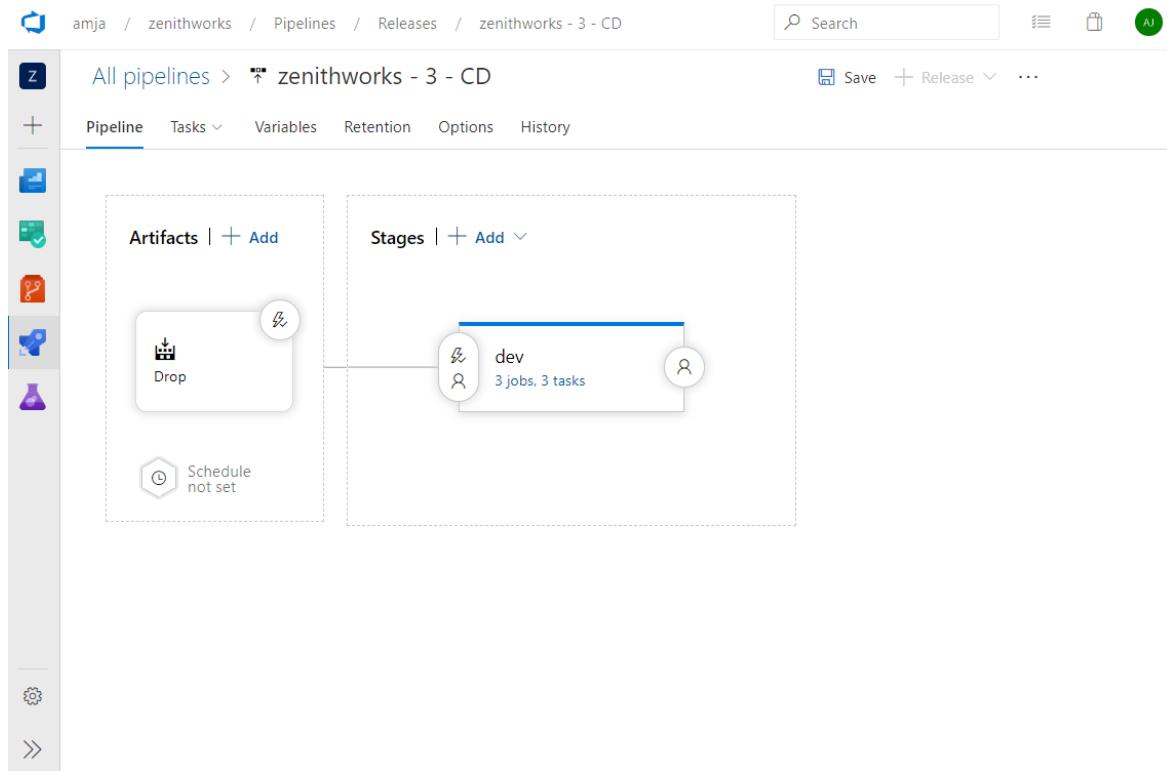
6. Select the Deployment strategy dropdown menu, and then select **Blue-Green**.

The screenshot shows the Azure portal interface for configuring continuous delivery. On the left, there's a sidebar with various settings like Overview, Activity log, and Tags. The main area is titled 'Continuous delivery' and contains instructions about setting up a release pipeline. A prominent 'Configure' button is at the bottom. To the right, a detailed configuration dialog is open. It asks for the Azure DevOps Organization (set to 'amja' and 'Use existing'). The 'Project' is set to 'zenithworks'. Under 'Deployment group', 'Use existing' is selected. The 'Deployment group name' is 'production-dg'. The 'Build pipeline' is 'zw-core-CI'. In the 'Deployment strategy' section, 'Blue-Green' is selected from a dropdown menu. At the bottom right of the dialog is a blue 'OK' button.

7. Add a "blue" or "green" tag to VMs that are used for blue-green deployments. If a VM is for a standby role, tag it as "green". Otherwise, tag it as "blue".

This screenshot is similar to the previous one but includes a 'Tags' section at the bottom of the configuration dialog. This section contains a single tag labeled 'green'. The rest of the dialog is identical to the first screenshot, showing the 'Deployment strategy' dropdown with 'Blue-Green' selected.

8. Select **OK** to configure the classic release pipeline to deploy to your virtual machine.



9. Navigate to your release pipeline and then select **Edit** to view the pipeline configuration. In this example, the *dev* stage is composed of three jobs:
  - a. Deploy Green: the app is deployed to a standby VM tagged "green".
  - b. Wait for manual resumption: the pipeline pauses and waits for manual intervention.
  - c. Swap Blue-Green: this job swaps the "blue" and "green" tags in the VMs. This ensures that VMs with older application versions are now tagged as "green". During the next pipeline run, applications will be deployed to these VMs.

The screenshot shows the detailed configuration of the 'dev' stage in the Azure DevOps Pipeline. The stage contains the following tasks:

- Deployment group job**: Deployment name is 'Deploy Green'.
- Deployment targets**: Deployment group is 'production-dg'.
- Required tags**: Tag is 'green'.
- Targets to deploy in parallel**: Multiple targets are selected.
- Maximum number of targets in parallel**: Set to 100%.
- Timeout**: Set to 10 minutes.

## Resources

- [Deploy to Azure virtual machines with Azure DevOps](#)
- [Deploy to an Azure virtual machine scale set](#)

## Related articles

- [Configure the rolling deployment strategy](#)
- [Configure the canary deployment strategy](#)

# Deploy your app to Linux virtual machines in Azure using Azure DevOps Services and Azure Pipelines

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

Continuous integration (CI) and continuous deployment (CD) form a pipeline by which you can build, release, and deploy your code after every code commit. This document contains the steps associated with setting up a CI/CD pipeline for doing multi-machine deployments using Azure Pipelines.

Azure Pipelines provides a complete, fully featured set of CI/CD automation tools for deployments to Virtual machines, both on-prem or on any cloud.

In this tutorial, you will set up a YAML based CI/CD pipeline to deploy your app to an Azure Pipelines [Environment](#) with Linux Virtual machines as resources, each of which serve as web servers to run the app.

You learn how to:

- Get a sample app.
- Create a YAML based Azure Pipelines CI pipeline for building the sample app.
- Create an Azure Pipelines Environment for the Azure virtual machines
- Create an Azure Pipelines CD pipeline.
- Execute manual and CI-triggered deployments.

## Before you begin

- Sign in to your Azure DevOps Services organization (<https://dev.azure.com/>). You can get a free [Azure DevOps Services organization](#).

### NOTE

For more information, see [Connect to Azure DevOps Services](#).

- You need a Linux virtual machine for a deployment target. For more information, see [Create and manage Linux VMs with the Azure CLI](#).
- Open inbound port 80 for your virtual machine. For more information, see [Create network security groups using the Azure portal](#).

## Get your sample app code

If you already have an app in GitHub that you want to deploy, you can try creating a pipeline for that code.

However, if you are a new user, then you might get a better start by using our sample code. In that case, fork this repo in GitHub:

- [Java](#)
- [JavaScript](#)

<https://github.com/spring-projects/spring-petclinic>

#### NOTE

Petclinic is a [Java Spring Boot](#) application built using [Maven](#).

## Prerequisites for the Linux VM

Sample apps mentioned above have been tested on Ubuntu 16.04, and we recommend you use the same version of Linux VM for this quickstart. Follow the additional steps described below based on the runtime stack used for the app.

- [Java](#)
- [JavaScript](#)
- For deploying Java Spring Boot and Spring Cloud based apps, create a Linux VM in Azure using [this template](#), which provides a fully supported OpenJDK-based runtime.
- For deploying Java servlets on Tomcat server, create a Linux VM with Java 8 using [this Azure template](#) and [configure Tomcat 9.x as a service](#).
- For deploying Java EE based app, use an Azure template to create a [Linux VM + Java + WebSphere 9.x](#) or a [Linux VM + Java + WebLogic 12.x](#) or a [Linux VM + Java](#) + WildFly/JBoss 14

## Create an Azure Pipelines environment with Azure virtual machines

Virtual machines can be added as resources within [environments](#) and can be targeted for multi-machine deployments. Deployment history views within environment provide traceability from VM to the pipeline and then to the commit.

You can create an environment in the “[Environments](#)” hub within the “[Pipelines](#)” section.

1. Sign in to your Azure DevOps organization and navigate to your project.
2. In your project, navigate to the [Pipelines](#) page. Then choose [Environments](#) and click [Create Environment](#). Specify a **Name** (required) for the environment and a **Description**.
3. Choose **Virtual Machines** as a **Resource** to be added to the environment and click **Next**.
4. Choose Operating System (Windows/Linux), and **copy PS registration script**.
5. Now run the copied script from an administrator PowerShell command prompt on each of the target VMs to be registered with this Environment.

#### NOTE

- Personal Access Token of the logged in user is pre-inserted in the script which expires on the same day making the copied script unusable thereon.
- If your VM already has any agent running on it, provide a unique name for “agent” to register with environment.

6. Once the VM is registered, it will start appearing as an environment resource under “resources” tab of the environment.

## New environment

X

### Virtual machine resource

#### Provider

Generic provider

#### Operating system

Windows

#### Registration script

1. Run `$ErrorActionPreference="Stop"` in administrative powershell

7. For adding more VMs, you can view and copy the script again by clicking on "Add resource" and choosing "Virtual Machines" as resource. This script would remain same for all the VMs to be added to this environment.
8. Each machine interacts with Azure Pipelines to coordinate deployment of your app.

The screenshot shows the 'Resources' tab for the 'VMenv' environment. There is one item listed:

Name	Latest job
USHAN-PC	Never deployed

9. You can add tags to the VM as part of the interactive PowerShell registration script (or) you can also add/remove the same from the resource view by clicking on the triple dots at the end of each VM resource in the resources view.

The tags you assign allow you to limit deployment to specific virtual machines when the environment is used in a Deployment job. Tags are each limited to 256 characters, but there is no limit to the number of tags you can use.

The screenshot shows the 'Resources' tab for the 'VMenv' environment. A modal window titled 'Manage tags for USHAN-PC' is open, displaying the tag 'web'. The modal has 'Cancel' and 'Save' buttons.

## Define your CI build pipeline

You'll need a continuous integration (CI) build pipeline that publishes your web application, as well as a deployment script that can be run locally on the Ubuntu server. Set up a CI build pipeline based on the runtime you want to use.

1. Sign in to your Azure DevOps organization and navigate to your project.
2. In your project, navigate to the **Pipelines** page. Then choose the action to create a new pipeline.
3. Walk through the steps of the wizard by first selecting **GitHub** as the location of your source code.
4. You might be redirected to GitHub to sign in. If so, enter your GitHub credentials.
5. When the list of repositories appears, select your desired sample app repository.
6. Azure Pipelines will analyze your repository and recommend a suitable pipeline template.
  - [Java](#)
  - [JavaScript](#)

Select the **starter** template and copy the below YAML snippet that builds your Java project and runs tests with Apache Maven:

```
jobs:  
- job: Build  
  displayName: Build Maven Project  
  steps:  
  - task: Maven@3  
    displayName: 'Maven Package'  
    inputs:  
      mavenPomFile: 'pom.xml'  
  - task: CopyFiles@2  
    displayName: 'Copy Files to artifact staging directory'  
    inputs:  
      SourceFolder: '$(System.DefaultWorkingDirectory)'  
      Contents: '**/target/*.{war|jar}'  
      TargetFolder: '$(Build.ArtifactStagingDirectory)'  
  - upload: '$(Build.ArtifactStagingDirectory)'  
    artifact: drop
```

For more guidance, follow the steps mentioned in [Build your Java app with Maven](#).

## Define CD steps to deploy to the Linux VM

1. Change the YAML file for the above pipeline to include a [deployment job](#) by referencing the environment and the VM resources which you have earlier using the YAML syntax below:

```
jobs:  
- deployment: VMDeploy  
  displayName: web  
  environment:  
    name: <environment name>  
    resourceType: VirtualMachine  
    tags: web
```

2. You can select specific sets of virtual machines from the environment to receive the deployment by specifying the **tags** that you have defined for each virtual machine in the environment. [Here](#) is the complete YAML schema for Deployment job.

3. You can specify either `runOnce` or `rolling` as deployment strategy.

`runOnce` is the simplest deployment strategy wherein all the life cycle hooks, namely `preDeploy`, `deploy`, `routeTraffic`, and `postRouteTraffic`, are executed once. Then, either `on: success` or `on: failure` is executed.

Below is the example YAML snippet for `runOnce`:

```
jobs:
- deployment: VMDeploy
  displayName: web
  pool:
    vmImage: 'Ubuntu-latest'
  environment:
    name: <environment name>
    resourceType: VirtualMachine
  strategy:
    runOnce:
      deploy:
        steps:
          - script: echo my first deployment
```

4. Below is an example of the YAML snippet that you can use to define a rolling strategy for Virtual machines updates upto 5 targets in each iteration. `maxParallel` will determine the number of targets that can be deployed to, in parallel. The selection accounts for absolute number or percentage of targets that must remain available at any time excluding the targets that are being deployed to. It is also used to determine the success and failure conditions during deployment.

```

jobs:
- deployment: VMDeploy
  displayName: web
  environment:
    name: <environment name>
    resourceType: VirtualMachine
  strategy:
    rolling:
      maxParallel: 5 #for percentages, mention as x%
      preDeploy:
        steps:
          - download: current
            artifact: drop
          - script: echo initialize, cleanup, backup, install certs
    deploy:
      steps:
        - task: Bash@3
          inputs:
            targetType: 'inline'
            script: |
              # Modify deployment script based on the app type
              echo "Starting deployment script run"
              sudo java -jar '$(Pipeline.Workspace)/drop/**/target/*.jar'
  routeTraffic:
    steps:
      - script: echo routing traffic
  postRouteTraffic:
    steps:
      - script: echo health check post-route traffic
  on:
    failure:
      steps:
        - script: echo Restore from backup! This is on failure
    success:
      steps:
        - script: echo Notify! This is on success

```

With each run of this job, deployment history is recorded against the <environment name> environment that you have created and registered the VMs.

## Run your pipeline and get traceability views in environment

Deployments view of the environment provides complete traceability of commits and work items, and a cross-pipeline deployment history per environment/resource.

Run	Jobs	
Update rolling-deployment.yml for Azure Pipelines #20191209.1 on niadak.AspNetCore	VMDeploy	Monday <1s
Update rolling-deployment.yml for Azure Pipelines #20191206.1 on niadak.AspNetCore	VMDeploy	Friday <1s
Update rolling-deployment.yml for Azure Pipelines #20191205.1 on niadak.AspNetCore	VMDeploy	Thursday <1s

← Deployment by 20191205.1  
#2016 on niadak.AspNetCore targeting Niadak-VM-Env

Jobs Changes Workitems

Jobs	
web_VM01_PreDeploy	Thursday ⌚ 12s
web_VM01_Deploy	Thursday ⌚ 15s
web_VM01_RouteTraffic	Thursday ⌚ 13s
web_VM01_PostRouteTraffic	Thursday ⌚ 11s
web_VM01_OnSuccess	Thursday ⌚ 11s

## Next steps

- You can proceed to [customize the pipeline](#) you just created.
- To learn what else you can do in YAML pipelines, see [YAML schema reference](#).
- To learn about how to deploy a LAMP (Linux, Apache, MySQL, and PHP) stack, advance to the next tutorial.

[Deploy LAMP stack](#)

# Tutorial: Use TLS/SSL certificates to secure a web server

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

To secure web servers, a Transport Layer Security (TLS), previously known as Secure Sockets Layer (SSL), certificate can be used to encrypt web traffic. These TLS/SSL certificates can be stored in Azure Key Vault, and allow secure deployments of certificates to Linux virtual machines (VMs) in Azure. In this tutorial you learn how to:

- Create an Azure Key Vault
- Generate or upload a certificate to the Key Vault
- Create a VM and install the NGINX web server
- Inject the certificate into the VM and configure NGINX with a TLS binding

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Overview

Azure Key Vault safeguards cryptographic keys and secrets, such as certificates or passwords. Key Vault helps streamline the certificate management process and enables you to maintain control of keys that access those certificates. You can create a self-signed certificate inside Key Vault, or upload an existing, trusted certificate that you already own.

Rather than using a custom VM image that includes certificates baked-in, you inject certificates into a running VM. This process ensures that the most up-to-date certificates are installed on a web server during deployment. If you renew or replace a certificate, you don't also have to create a new custom VM image. The latest certificates are automatically injected as you create additional VMs. During the whole process, the certificates never leave the Azure platform or are exposed in a script, command-line history, or template.

## Create an Azure Key Vault

Before you can create a Key Vault and certificates, create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroupSecureWeb* in the *eastus* location:

```
az group create --name myResourceGroupSecureWeb --location eastus
```

Next, create a Key Vault with [az keyvault create](#) and enable it for use when you deploy a VM. Each Key Vault requires a unique name, and should be all lowercase. Replace *<mykeyvault>* in the following example with your own unique Key Vault name:

```
keyvault_name=<mykeyvault>
az keyvault create \
--resource-group myResourceGroupSecureWeb \
--name $keyvault_name \
--enabled-for-deployment
```

## Generate a certificate and store in Key Vault

For production use, you should import a valid certificate signed by trusted provider with [az keyvault certificate import](#). For this tutorial, the following example shows how you can generate a self-signed certificate with [az keyvault certificate create](#) that uses the default certificate policy:

```
az keyvault certificate create \
--vault-name $keyvault_name \
--name mycert \
--policy "$(az keyvault certificate get-default-policy)"
```

### Prepare a certificate for use with a VM

To use the certificate during the VM create process, obtain the ID of your certificate with [az keyvault secret list-versions](#). Convert the certificate with [az vm secret format](#). The following example assigns the output of these commands to variables for ease of use in the next steps:

```
secret=$(az keyvault secret list-versions \
--vault-name $keyvault_name \
--name mycert \
--query "[?attributes.enabled].id" --output tsv)
vm_secret=$(az vm secret format --secrets "$secret" -g myResourceGroupSecureWeb --keyvault $keyvault_name)
```

### Create a cloud-init config to secure NGINX

[Cloud-init](#) is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files, or to configure users and security. As cloud-init runs during the initial boot process, there are no additional steps or required agents to apply your configuration.

When you create a VM, certificates and keys are stored in the protected `/var/lib/waagent/` directory. To automate adding the certificate to the VM and configuring the web server, use cloud-init. In this example, you install and configure the NGINX web server. You can use the same process to install and configure Apache.

Create a file named `cloud-init-web-server.txt` and paste the following configuration:

```

#cloud-config
package_upgrade: true
packages:
- nginx
write_files:
- owner: www-data:www-data
- path: /etc/nginx/sites-available/default
content: |
  server {
    listen 443 ssl;
    ssl_certificate /etc/nginx/ssl/mycert.cert;
    ssl_certificate_key /etc/nginx/ssl/mycert.prv;
  }
runcmd:
- secretsname=$(find /var/lib/waagent/ -name "*.prv" | cut -c -57)
- mkdir /etc/nginx/ssl
- cp $secretsname.crt /etc/nginx/ssl/mycert.cert
- cp $secretsname.prv /etc/nginx/ssl/mycert.prv
- service nginx restart

```

## Create a secure VM

Now create a VM with [az vm create](#). The certificate data is injected from Key Vault with the `--secrets` parameter. You pass in the cloud-init config with the `--custom-data` parameter:

```

az vm create \
--resource-group myResourceGroupSecureWeb \
--name myVM \
--image UbuntuLTS \
--admin-username azureuser \
--generate-ssh-keys \
--custom-data cloud-init-web-server.txt \
--secrets "$vm_secret"

```

It takes a few minutes for the VM to be created, the packages to install, and the app to start. When the VM has been created, take note of the `publicIpAddress` displayed by the Azure CLI. This address is used to access your site in a web browser.

To allow secure web traffic to reach your VM, open port 443 from the Internet with [az vm open-port](#):

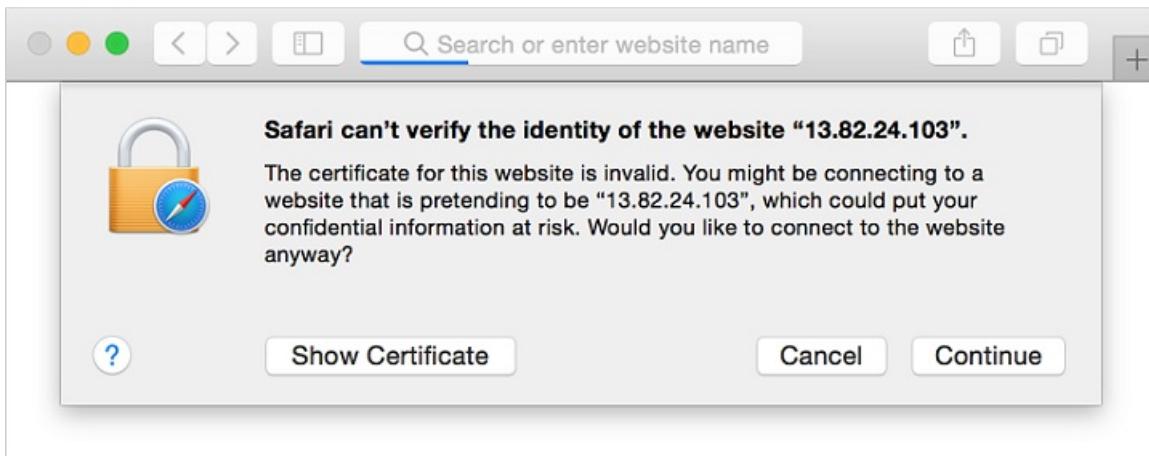
```

az vm open-port \
--resource-group myResourceGroupSecureWeb \
--name myVM \
--port 443

```

## Test the secure web app

Now you can open a web browser and enter `https://<publicIpAddress>` in the address bar. Provide your own public IP address from the VM create process. Accept the security warning if you used a self-signed certificate:



Your secured NGINX site is then displayed as in the following example:



## Next steps

In this tutorial, you secured an NGINX web server with a TLS/SSL certificate stored in Azure Key Vault. You learned how to:

- Create an Azure Key Vault
- Generate or upload a certificate to the Key Vault
- Create a VM and install the NGINX web server
- Inject the certificate into the VM and configure NGINX with a TLS binding

Follow this link to see pre-built virtual machine script samples.

[Linux virtual machine script samples](#)

# Create a VM from a VHD by using the Azure portal

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

There are several ways to create a virtual machine (VM) in Azure:

- If you already have a virtual hard disk (VHD) to use or you want to copy the VHD from an existing VM to use, you can create a new VM by *attaching* the VHD to the new VM as an OS disk.
- You can create a new VM from the VHD of a VM that has been deleted. For example, if you have an Azure VM that isn't working correctly, you can delete the VM and use its VHD to create a new VM. You can either reuse the same VHD or create a copy of the VHD by creating a snapshot and then creating a new managed disk from the snapshot. Although creating a snapshot takes a few more steps, it preserves the original VHD and provides you with a fallback.
- Take a classic VM and use the VHD to create a new VM that uses the Resource Manager deployment model and managed disks. For the best results, **Stop** the classic VM in the Azure portal before creating the snapshot.
- You can create an Azure VM from an on-premises VHD by uploading the on-premises VHD and attaching it to a new VM. You use PowerShell or another tool to upload the VHD to a storage account, and then you create a managed disk from the VHD. For more information, see [Upload a specialized VHD](#).

## IMPORTANT

When you use a specialized disk to create a new VM, the new VM retains the computer name of the original VM. Other computer-specific information (e.g. CMID) is also kept and, in some cases, this duplicate information could cause issues. When copying a VM, be aware of what types of computer-specific information your applications rely on. Thus, don't use a specialized disk if you want to create multiple VMs. Instead, for larger deployments, [create an image](#) and then [use that image to create multiple VMs](#).

We recommend that you limit the number of concurrent deployments to 20 VMs from a single snapshot or VHD.

## Copy a disk

Create a snapshot and then create a disk from the snapshot. This strategy allows you to keep the original VHD as a fallback:

1. From the [Azure portal](#), on the left menu, select **All services**.
2. In the **All services** search box, enter **disks** and then select **Disk** to display the list of available disks.
3. Select the disk that you would like to use. The **Disk** page for that disk appears.
4. From the menu at the top, select **Create snapshot**.
5. Choose a **Resource group** for the snapshot. You can use either an existing resource group or create a new one.
6. Enter a **Name** for the snapshot.
7. For **Snapshot type**, choose either **Full** or **Incremental**.
8. For **Storage type**, choose **Standard HDD**, **Premium SSD**, or **Zone-redundant storage**.
9. When you're done, select **Create** to create the snapshot.

10. After the snapshot has been created, select **Create a resource** in the left menu.
11. In the search box, enter **managed disk** and then select **Managed Disks** from the list.
12. On the **Managed Disks** page, select **Create**.
13. Choose a **Resource group** for the disk. You can use either an existing resource group or create a new one.  
This selection will also be used as the resource group where you create the VM from the disk.
14. Enter a **Name** for the disk.
15. In **Source type**, ensure **Snapshot** is selected.
16. In the **Source snapshot** drop-down, select the snapshot you want to use.
17. For **Size**, choose either **Standard (HDD)** or **Premium (SSD)** storage.
18. Make any other adjustments as needed and then select **Create** to create the disk.

## Create a VM from a disk

After you have the managed disk VHD that you want to use, you can create the VM in the portal:

1. From the [Azure portal](#), on the left menu, select **All services**.
2. In the **All services** search box, enter **disks** and then select **Disks** to display the list of available disks.
3. Select the disk that you would like to use. The **Disk** page for that disk opens.
4. In the **Overview** page, ensure that **DISK STATE** is listed as **Unattached**. If it isn't, you might need to either detach the disk from the VM or delete the VM to free up the disk.
5. In the menu at the top of the page, select **Create VM**.
6. On the **Basics** page for the new VM, enter a **Virtual machine name** and either select an existing **Resource group** or create a new one.
7. For **Size**, select **Change size** to access the **Size** page.
8. Select a VM size row and then choose **Select**.
9. On the **Disks** page, you may notice that the "OS Disk Type" cannot be changed. This preselected value is configured at the point of Snapshot or VHD creation and will carry over to the new VM. If you need to modify disk type take a new snapshot from an existing VM or disk.
10. On the **Networking** page, you can either let the portal create all new resources or you can select an existing **Virtual network** and **Network security group**. The portal always creates a new network interface and public IP address for the new VM.
11. On the **Management** page, make any changes to the monitoring options.
12. On the **Guest config** page, add any extensions as needed.
13. When you're done, select **Review + create**.
14. If the VM configuration passes validation, select **Create** to start the deployment.

## Next steps

You can also use PowerShell to [upload a VHD to Azure and create a specialized VM](#).

# Create a Windows VM from a specialized disk by using PowerShell

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Create a new VM by attaching a specialized managed disk as the OS disk. A specialized disk is a copy of a virtual hard disk (VHD) from an existing VM that contains the user accounts, applications, and other state data from your original VM.

You have several options:

- [Use an existing managed disk](#). This option is useful if you have a VM that isn't working correctly. You can delete the VM and then reuse the managed disk to create a new VM.
- [Upload a VHD](#)
- [Copy an existing Azure VM by using snapshots](#)

You can also use the Azure portal to [create a new VM from a specialized VHD](#).

This article shows you how to use managed disks. If you have a legacy deployment that requires using a storage account, see [Create a VM from a specialized VHD in a storage account](#).

## IMPORTANT

When you use a specialized disk to create a new VM, the new VM retains the computer name of the original VM.

Other computer-specific information (e.g. CMID) is also kept and, in some cases, this duplicate information could cause issues.

When copying a VM, be aware of what types of computer-specific information your applications rely on.

Thus, don't use a specialized disk if you want to create multiple VMs. Instead, for larger deployments, [create an image](#) and then [use that image to create multiple VMs](#).

We recommend that you limit the number of concurrent deployments to 20 VMs from a single VHD or snapshot.

## Option 1: Use an existing disk

If you had a VM that you deleted and you want to reuse the OS disk to create a new VM, use [Get-AzDisk](#).

```
$resourceGroupName = 'myResourceGroup'  
$osDiskName = 'myOsDisk'  
$osDisk = Get-AzDisk `  
-ResourceGroupName $resourceGroupName `  
-DiskName $osDiskName
```

You can now attach this disk as the OS disk to a [new VM](#).

## Option 2: Upload a specialized VHD

You can upload the VHD from a specialized VM created with an on-premises virtualization tool, like Hyper-V, or a VM exported from another cloud.

### Prepare the VM

Use the VHD as-is to create a new VM.

- [Prepare a Windows VHD to upload to Azure](#). Do not generalize the VM by using Sysprep.
- Remove any guest virtualization tools and agents that are installed on the VM (such as VMware tools).
- Make sure the VM is configured to get the IP address and DNS settings from DHCP. This ensures that the server obtains an IP address within the virtual network when it starts up.

## Upload the VHD

You can now upload a VHD straight into a managed disk. For instructions, see [Upload a VHD to Azure using Azure PowerShell](#).

## Option 3: Copy an existing Azure VM

You can create a copy of a VM that uses managed disks by taking a snapshot of the VM, and then by using that snapshot to create a new managed disk and a new VM.

If you want to copy an existing VM to another region, you might want to use azcopy to [create a copy of a disk in another region](#).

### Take a snapshot of the OS disk

You can take a snapshot of an entire VM (including all disks) or of just a single disk. The following steps show you how to take a snapshot of just the OS disk of your VM with the [New-AzSnapshot cmdlet](#).

First, set some parameters.

```
$resourceGroupName = 'myResourceGroup'  
$vmName = 'myVM'  
$location = 'westus'  
$snapshotName = 'mySnapshot'
```

Get the VM object.

```
$vm = Get-AzVM -Name $vmName `  
-ResourceGroupName $resourceGroupName
```

Get the OS disk name.

```
$disk = Get-AzDisk -ResourceGroupName $resourceGroupName `  
-DiskName $vm.StorageProfile.OsDisk.Name
```

Create the snapshot configuration.

```
$snapshotConfig = New-AzSnapshotConfig `  
-SourceUri $disk.Id `  
-OsType Windows `  
-CreateOption Copy `  
-Location $location
```

Take the snapshot.

```
$snapShot = New-AzSnapshot `  
-Snapshot $snapshotConfig `  
-SnapshotName $snapshotName `  
-ResourceGroupName $resourceGroupName
```

To use this snapshot to create a VM that needs to be high-performing, add the parameter `-AccountType Premium_LRS` to the `New-AzSnapshotConfig` command. This parameter creates the snapshot so that it's stored as a Premium Managed Disk. Premium Managed Disks are more expensive than Standard, so be sure you'll need Premium before using this parameter.

### Create a new disk from the snapshot

Create a managed disk from the snapshot by using [New-AzDisk](#). This example uses *myOSDisk* for the disk name.

Create a new resource group for the new VM.

```
$destinationResourceGroup = 'myDestinationResourceGroup'  
New-AzResourceGroup -Location $location  
-Name $destinationResourceGroup
```

Set the OS disk name.

```
$osDiskName = 'myOsDisk'
```

Create the managed disk.

```
$osDisk = New-AzDisk -DiskName $osDiskName -Disk `  
    (New-AzDiskConfig -Location $location -CreateOption Copy `  
        -SourceResourceId $snapshot.Id) `  
    -ResourceGroupName $destinationResourceGroup
```

## Create the new VM

Create networking and other VM resources to be used by the new VM.

### Create the subnet and virtual network

Create the [virtual network](#) and subnet for the VM.

1. Create the subnet. This example creates a subnet named *mySubNet*, in the resource group *myDestinationResourceGroup*, and sets the subnet address prefix to *10.0.0.0/24*.

```
$subnetName = 'mySubNet'  
$singleSubnet = New-AzVirtualNetworkSubnetConfig `  
    -Name $subnetName `  
    -AddressPrefix 10.0.0.0/24
```

2. Create the virtual network. This example sets the virtual network name to *myVnetName*, the location to *West US*, and the address prefix for the virtual network to *10.0.0.0/16*.

```
$vnetName = "myVnetName"  
$vnet = New-AzVirtualNetwork `  
    -Name $vnetName -ResourceGroupName $destinationResourceGroup `  
    -Location $location `  
    -AddressPrefix 10.0.0.0/16 `  
    -Subnet $singleSubnet
```

### Create the network security group and an RDP rule

To be able to sign in to your VM with remote desktop protocol (RDP), you'll need to have a security rule that allows RDP access on port 3389. In our example, the VHD for the new VM was created from an existing

specialized VM, so you can use an account that existed on the source virtual machine for RDP. This example denies RDP traffic, to be more secure. You can change `-Access` to `Allow` if you want to allow RDP access.

This example sets the network security group (NSG) name to *myNsg* and the RDP rule name to *myRdpRule*.

```
$nsgName = "myNsg"

$rdpRule = New-AzNetworkSecurityRuleConfig -Name myRdpRule -Description "Deny RDP" ` 
    -Access Deny -Protocol Tcp -Direction Inbound -Priority 110 ` 
    -SourceAddressPrefix Internet -SourcePortRange * ` 
    -DestinationAddressPrefix * -DestinationPortRange 3389
$nsg = New-AzNetworkSecurityGroup ` 
    -ResourceGroupName $destinationResourceGroup ` 
    -Location $location ` 
    -Name $nsgName -SecurityRules $rdpRule
```

For more information about endpoints and NSG rules, see [Opening ports to a VM in Azure by using PowerShell](#).

### Create a public IP address and NIC

To enable communication with the virtual machine in the virtual network, you'll need a [public IP address](#) and a network interface.

1. Create the public IP. In this example, the public IP address name is set to *myIP*.

```
$ipName = "myIP"
$pip = New-AzPublicIpAddress ` 
    -Name $ipName -ResourceGroupName $destinationResourceGroup ` 
    -Location $location ` 
    -AllocationMethod Dynamic
```

2. Create the NIC. In this example, the NIC name is set to *myNicName*.

```
$nicName = "myNicName"
$nic = New-AzNetworkInterface -Name $nicName ` 
    -ResourceGroupName $destinationResourceGroup ` 
    -Location $location -SubnetId $vnet.Subnets[0].Id ` 
    -PublicIpAddressId $pip.Id ` 
    -NetworkSecurityGroupId $nsg.Id
```

### Set the VM name and size

This example sets the VM name to *myVM* and the VM size to *Standard\_A2*.

```
$vmName = "myVM"
$vmConfig = New-AzVMConfig -VMName $vmName -VMSize "Standard_A2"
```

### Add the NIC

```
$vm = Add-AzVMNetworkInterface -VM $vmConfig -Id $nic.Id
```

### Add the OS disk

Add the OS disk to the configuration by using [Set-AzVMOSDisk](#). This example sets the size of the disk to *128 GB* and attaches the managed disk as a *Windows* OS disk.

```
$vm = Set-AzVMOSDisk -VM $vm -ManagedDiskId $osDisk.Id -StorageAccountType Standard_LRS `  
-DiskSizeInGB 128 -CreateOption Attach -Windows
```

## Complete the VM

Create the VM by using [New-AzVM](#) with the configurations that we just created.

```
New-AzVM -ResourceGroupName $destinationResourceGroup -Location $location -VM $vm
```

If this command is successful, you'll see output like this:

```
RequestId IsSuccessStatusCode StatusCode ReasonPhrase  
-----  
True      OK    OK
```

## Verify that the VM was created

You should see the newly created VM either in the [Azure portal](#) under **Browse > Virtual machines**, or by using the following PowerShell commands.

```
$vmList = Get-AzVM -ResourceGroupName $destinationResourceGroup  
$vmList.Name
```

## Next steps

Sign in to your new virtual machine. For more information, see [How to connect and log on to an Azure virtual machine running Windows](#).

# Create a Windows virtual machine from a Resource Manager template

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Learn how to create a Windows virtual machine by using an Azure Resource Manager template and Azure PowerShell from the Azure Cloud shell. The template used in this article deploys a single virtual machine running Windows Server in a new virtual network with a single subnet. For creating a Linux virtual machine, see [How to create a Linux virtual machine with Azure Resource Manager templates](#).

An alternative is to deploy the template from the Azure portal. To open the template in the portal, select the **Deploy to Azure** button.



## Create a virtual machine

Creating an Azure virtual machine usually includes two steps:

- Create a resource group. An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine.
- Create a virtual machine.

The following example creates an [Azure Generation 2 VM](#) by default from an [Azure Quickstart template](#). Here is a copy of the template:

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "metadata": {  
    "_generator": {  
      "name": "bicep",  
      "version": "0.8.9.13224",  
      "templateHash": "15495738823141086515"  
    }  
  },  
  "parameters": {  
    "adminUsername": {  
      "type": "string",  
      "metadata": {  
        "description": "Username for the Virtual Machine."  
      }  
    },  
    "adminPassword": {  
      "type": "secureString",  
      "minLength": 12,  
      "metadata": {  
        "description": "Password for the Virtual Machine."  
      }  
    },  
    "dnsLabelPrefix": {  
      "type": "string",  
      "defaultValue": "[toLower(format('{0}-{1}', parameters('vmName'), uniqueString(resourceGroup().id, parameters('vmName'))))]",  
      "metadata": {  
        "description": "The DNS label for the virtual machine, used for the fully qualified domain name (FQDN). It is generated by concatenating the vmName parameter with a unique string based on the resource group ID."  
      }  
    }  
  }  
}
```

```
        "description": "Unique DNS Name for the Public IP used to access the Virtual Machine."
    }
},
"publicIpName": {
    "type": "string",
    "defaultValue": "myPublicIP",
    "metadata": {
        "description": "Name for the Public IP used to access the Virtual Machine."
    }
},
"publicIPAllocationMethod": {
    "type": "string",
    "defaultValue": "Dynamic",
    "allowedValues": [
        "Dynamic",
        "Static"
    ],
    "metadata": {
        "description": "Allocation method for the Public IP used to access the Virtual Machine."
    }
},
"publicIpsku": {
    "type": "string",
    "defaultValue": "Basic",
    "allowedValues": [
        "Basic",
        "Standard"
    ],
    "metadata": {
        "description": "SKU for the Public IP used to access the Virtual Machine."
    }
},
"OSVersion": {
    "type": "string",
    "defaultValue": "2022-datacenter-azure-edition-core",
    "allowedValues": [
        "2008-R2-SP1",
        "2008-R2-SP1-smalldisk",
        "2012-Datacenter",
        "2012-datacenter-gensecond",
        "2012-Datacenter-smalldisk",
        "2012-datacenter-smalldisk-g2",
        "2012-Datacenter-zhcn",
        "2012-datacenter-zhcn-g2",
        "2012-R2-Datacenter",
        "2012-r2-datacenter-gensecond",
        "2012-R2-Datacenter-smalldisk",
        "2012-r2-datacenter-smalldisk-g2",
        "2012-R2-Datacenter-zhcn",
        "2012-r2-datacenter-zhcn-g2",
        "2016-Datacenter",
        "2016-datacenter-gensecond",
        "2016-datacenter-gs",
        "2016-Datacenter-Server-Core",
        "2016-datacenter-server-core-g2",
        "2016-Datacenter-Server-Core-smalldisk",
        "2016-datacenter-server-core-smalldisk-g2",
        "2016-Datacenter-smalldisk",
        "2016-datacenter-smalldisk-g2",
        "2016-Datacenter-with-Containers",
        "2016-datacenter-with-containers-g2",
        "2016-datacenter-with-containers-gs",
        "2016-Datacenter-zhcn",
        "2016-datacenter-zhcn-g2",
        "2019-Datacenter",
        "2019-Datacenter-Core",
        "2019-datacenter-core-g2",
        "2019-Datacenter-Core-smalldisk",
        "2019-datacenter-core-smalldisk-g2",
        "2019-Datacenter-Container-Optimized"
    ]
}
```

```

        "2019-Datacenter-Core-with-Containers",
        "2019-datacenter-core-with-containers-g2",
        "2019-Datacenter-Core-with-Containers-smalldisk",
        "2019-datacenter-core-with-containers-smalldisk-g2",
        "2019-datacenter-gensecond",
        "2019-datacenter-gs",
        "2019-Datacenter-smalldisk",
        "2019-datacenter-smalldisk-g2",
        "2019-Datacenter-with-Containers",
        "2019-datacenter-with-containers-g2",
        "2019-datacenter-with-containers-gs",
        "2019-Datacenter-with-Containers-smalldisk",
        "2019-datacenter-with-containers-smalldisk-g2",
        "2019-Datacenter-zhcn",
        "2019-datacenter-zhcn-g2",
        "2022-datacenter",
        "2022-datacenter-azure-edition",
        "2022-datacenter-azure-edition-core",
        "2022-datacenter-azure-edition-core-smalldisk",
        "2022-datacenter-azure-edition-smalldisk",
        "2022-datacenter-core",
        "2022-datacenter-core-g2",
        "2022-datacenter-core-smalldisk",
        "2022-datacenter-core-smalldisk-g2",
        "2022-datacenter-g2",
        "2022-datacenter-smalldisk",
        "2022-datacenter-smalldisk-g2"
    ],
    "metadata": {
        "description": "The Windows version for the VM. This will pick a fully patched image of this given Windows version."
    }
},
"vmSize": {
    "type": "string",
    "defaultValue": "Standard_D2s_v5",
    "metadata": {
        "description": "Size of the virtual machine."
    }
},
"location": {
    "type": "string",
    "defaultValue": "[resourceGroup().location]",
    "metadata": {
        "description": "Location for all resources."
    }
},
"vmName": {
    "type": "string",
    "defaultValue": "simple-vm",
    "metadata": {
        "description": "Name of the virtual machine."
    }
}
},
"variables": {
    "storageAccountName": "[format('bootdiags{0}', uniqueString(resourceGroup().id))]",
    "nicName": "myVMNic",
    "addressPrefix": "10.0.0.0/16",
    "subnetName": "Subnet",
    "subnetPrefix": "10.0.0.0/24",
    "virtualNetworkName": "MyVNET",
    "networkSecurityGroupName": "default-NSG"
},
"resources": [
{
    "type": "Microsoft.Storage/storageAccounts",
    "apiVersion": "2021-04-01",
    "name": "[variables('storageAccountName')]"
}
]
}

```

```
"location": "[parameters('location')]",
"sku": {
    "name": "Standard_LRS"
},
"kind": "Storage"
},
{
    "type": "Microsoft.Network/publicIPAddresses",
    "apiVersion": "2021-02-01",
    "name": "[parameters('publicIpName')]",
    "location": "[parameters('location')]",
    "sku": {
        "name": "[parameters('publicIpSku')]"
    },
    "properties": {
        "publicIPAllocationMethod": "[parameters('publicIPAllocationMethod')]",
        "dnsSettings": {
            "domainNameLabel": "[parameters('dnsLabelPrefix')]"
        }
    }
},
{
    "type": "Microsoft.Network/networkSecurityGroups",
    "apiVersion": "2021-02-01",
    "name": "[variables('networkSecurityGroupName')]",
    "location": "[parameters('location')]",
    "properties": {
        "securityRules": [
            {
                "name": "default-allow-3389",
                "properties": {
                    "priority": 1000,
                    "access": "Allow",
                    "direction": "Inbound",
                    "destinationPortRange": "3389",
                    "protocol": "Tcp",
                    "sourcePortRange": "*",
                    "sourceAddressPrefix": "*",
                    "destinationAddressPrefix": "*"
                }
            }
        ]
    }
},
{
    "type": "Microsoft.Network/virtualNetworks",
    "apiVersion": "2021-02-01",
    "name": "[variables('virtualNetworkName')]",
    "location": "[parameters('location')]",
    "properties": {
        "addressSpace": {
            "addressPrefixes": [
                "[variables('addressPrefix')]"
            ]
        },
        "subnets": [
            {
                "name": "[variables('subnetName')]",
                "properties": {
                    "addressPrefix": "[variables('subnetPrefix')]",
                    "networkSecurityGroup": {
                        "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('networkSecurityGroupName'))]"
                    }
                }
            }
        ]
    },
    "dependsOn": [
```

```

        "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"
    ],
},
{
    "type": "Microsoft.Network/networkInterfaces",
    "apiVersion": "2021-02-01",
    "name": "[variables('nicName')]",
    "location": "[parameters('location')]",
    "properties": {
        "ipConfigurations": [
            {
                "name": "ipconfig1",
                "properties": {
                    "privateIPAllocationMethod": "Dynamic",
                    "publicIPAddress": {
                        "id": "[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]"
                    },
                    "subnet": {
                        "id": "[resourceId('Microsoft.Network/virtualNetworks/subnets',
variables('virtualNetworkName'), variables('subnetName'))]"
                    }
                }
            }
        ]
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/publicIPAddresses', parameters('publicIpName'))]",
        "[resourceId('Microsoft.Network/virtualNetworks', variables('virtualNetworkName'))]"
    ]
},
{
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2021-03-01",
    "name": "[parameters('vmName')]",
    "location": "[parameters('location')]",
    "properties": {
        "hardwareProfile": {
            "vmSize": "[parameters('vmSize')]"
        },
        "osProfile": {
            "computerName": "[parameters('vmName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "adminPassword": "[parameters('adminPassword')]"
        },
        "storageProfile": {
            "imageReference": {
                "publisher": "MicrosoftWindowsServer",
                "offer": "WindowsServer",
                "sku": "[parameters('OSVersion')]",
                "version": "latest"
            },
            "osDisk": {
                "createOption": "FromImage",
                "managedDisk": {
                    "storageAccountType": "StandardSSD_LRS"
                }
            },
            "dataDisks": [
                {
                    "diskSizeGB": 1023,
                    "lun": 0,
                    "createOption": "Empty"
                }
            ]
        },
        "networkProfile": {
            "networkInterfaces": [
                {
                    "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]"
                }
            ]
        }
    }
}

```

```

        }
    ],
},
"diagnosticsProfile": {
    "bootDiagnostics": {
        "enabled": true,
        "storageUri": "[reference(resourceId('Microsoft.Storage/storageAccounts',
variables('storageAccountName'))).primaryEndpoints.blob]"
    }
},
"dependsOn": [
    "[resourceId('Microsoft.Network/networkInterfaces', variables('nicName'))]",
    "[resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))]"
]
},
"outputs": {
    "hostname": {
        "type": "string",
        "value": "[reference(resourceId('Microsoft.Network/publicIPAddresses',
parameters('publicIpName'))).dnsSettings.fqdn]"
    }
}
}

```

To run the PowerShell script, Select **Try it** to open the Azure Cloud shell. To paste the script, right-click the shell, and then select **Paste**:

```

$resourceGroupName = Read-Host -Prompt "Enter the Resource Group name"
.setLocation = Read-Host -Prompt "Enter the location (i.e. centralus)"
$adminUsername = Read-Host -Prompt "Enter the administrator username"
$adminPassword = Read-Host -Prompt "Enter the administrator password" -AsSecureString
$dnsLabelPrefix = Read-Host -Prompt "Enter an unique DNS name for the public IP"

New-AzResourceGroup -Name $resourceGroupName -Location "$location"
New-AzResourceGroupDeployment ` 
    -ResourceGroupName $resourceGroupName ` 
    -TemplateUri "https://raw.githubusercontent.com/Azure/azure-quickstart-` 
templates/master/quickstarts/microsoft.compute/vm-simple-windows/azuredeploy.json" ` 
    -adminUsername $adminUsername ` 
    -adminPassword $adminPassword ` 
    -dnsLabelPrefix $dnsLabelPrefix

(Get-AzVm -ResourceGroupName $resourceGroupName).name

```

If you choose to install and use the PowerShell locally instead of from the Azure Cloud shell, this tutorial requires the Azure PowerShell module. Run `Get-Module -ListAvailable Az` to find the version. If you need to upgrade, see [Install Azure PowerShell module](#). If you're running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

In the previous example, you specified a template stored in GitHub. You can also download or create a template and specify the local path with the `--template-file` parameter.

Here are some additional resources:

- To learn how to develop Resource Manager templates, see [Azure Resource Manager documentation](#).
- To see the Azure virtual machine schemas, see [Azure template reference](#).
- To see more virtual machine template samples, see [Azure Quickstart templates](#).

## Connect to the virtual machine

The last PowerShell command from the previous script shows the virtual machine name. To connect to the virtual machine, see [How to connect and sign on to an Azure virtual machine running Windows](#).

## Next Steps

- If there were issues with the deployment, you might take a look at [Troubleshoot common Azure deployment errors with Azure Resource Manager](#).
- Learn how to create and manage a virtual machine in [Create and manage Windows VMs with the Azure PowerShell module](#).

To learn more about creating templates, view the JSON syntax and properties for the resources types you deployed:

- [Microsoft.Network/publicIPAddresses](#)
- [Microsoft.Network/virtualNetworks](#)
- [Microsoft.Network/networkInterfaces](#)
- [Microsoft.Compute/virtualMachines](#)

# Tutorial: Secure a web server on a Windows virtual machine in Azure with TLS/SSL certificates stored in Key Vault

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

## NOTE

Currently this doc only works for Generalized images. If attempting this tutorial using a Specialized disk you will receive an error.

To secure web servers, a Transport Layer Security (TLS), previously known as Secure Sockets Layer (SSL), certificate can be used to encrypt web traffic. These TLS/SSL certificates can be stored in Azure Key Vault, and allow secure deployments of certificates to Windows virtual machines (VMs) in Azure. In this tutorial you learn how to:

- Create an Azure Key Vault
- Generate or upload a certificate to the Key Vault
- Create a VM and install the IIS web server
- Inject the certificate into the VM and configure IIS with a TLS binding

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Overview

Azure Key Vault safeguards cryptographic keys and secrets, such certificates or passwords. Key Vault helps streamline the certificate management process and enables you to maintain control of keys that access those certificates. You can create a self-signed certificate inside Key Vault, or upload an existing, trusted certificate that you already own.

Rather than using a custom VM image that includes certificates baked-in, you inject certificates into a running VM. This process ensures that the most up-to-date certificates are installed on a web server during deployment. If you renew or replace a certificate, you don't also have to create a new custom VM image. The latest certificates are automatically injected as you create additional VMs. During the whole process, the certificates never leave the Azure platform or are exposed in a script, command-line history, or template.

## Create an Azure Key Vault

Before you can create a Key Vault and certificates, create a resource group with [New-AzResourceGroup](#). The following example creates a resource group named *myResourceGroupSecureWeb* in the *East US* location:

```
$resourceGroup = "myResourceGroupSecureWeb"
$location = "East US"
New-AzResourceGroup -ResourceGroupName $resourceGroup -Location $location
```

Next, create a Key Vault with [New-AzKeyVault](#). Each Key Vault requires a unique name, and should be all lower case. Replace `mykeyvault` in the following example with your own unique Key Vault name:

```
$keyvaultName="mykeyvault"
New-AzKeyVault -VaultName $keyvaultName ` 
    -ResourceGroup $resourceGroup ` 
    -Location $location ` 
    -EnabledForDeployment
```

## Generate a certificate and store in Key Vault

For production use, you should import a valid certificate signed by trusted provider with [Import-AzKeyVaultCertificate](#). For this tutorial, the following example shows how you can generate a self-signed certificate with [Add-AzKeyVaultCertificate](#) that uses the default certificate policy from [New-AzKeyVaultCertificatePolicy](#).

```
$policy = New-AzKeyVaultCertificatePolicy ` 
    -SubjectName "CN=www.contoso.com" ` 
    -SecretContentType "application/x-pkcs12" ` 
    -IssuerName Self ` 
    -ValidityInMonths 12

Add-AzKeyVaultCertificate ` 
    -VaultName $keyvaultName ` 
    -Name "mycert" ` 
    -CertificatePolicy $policy
```

## Create a virtual machine

Set an administrator username and password for the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Now you can create the VM with [New-AzVM](#). The following example creates a VM named *myVM* in the *EastUS* location. If they do not already exist, the supporting network resources are created. To allow secure web traffic, the cmdlet also opens port 443.

```

# Create a VM
New-AzVm `-
    -ResourceGroupName $resourceGroup `-
    -Name "myVM" `-
    -Location $location `-
    -VirtualNetworkName "myVnet" `-
    -SubnetName "mySubnet" `-
    -SecurityGroupName "myNetworkSecurityGroup" `-
    -PublicIpAddressName "myPublicIpAddress" `-
    -Credential $cred `-
    -OpenPorts 443

# Use the Custom Script Extension to install IIS
Set-AzVMExtension -ResourceGroupName $resourceGroup `-
    -ExtensionName "IIS" `-
    -VMName "myVM" `-
    -Location $location `-
    -Publisher "Microsoft.Compute" `-
    -ExtensionType "CustomScriptExtension" `-
    -TypeHandlerVersion 1.8 `-
    -SettingString '{"commandToExecute":"powershell Add-WindowsFeature Web-Server -IncludeManagementTools"}'

```

It takes a few minutes for the VM to be created. The last step uses the Azure Custom Script Extension to install the IIS web server with [Set-AzVmExtension](#).

## Add a certificate to VM from Key Vault

To add the certificate from Key Vault to a VM, obtain the ID of your certificate with [Get-AzKeyVaultSecret](#). Add the certificate to the VM with [Add-AzVMSecret](#):

```

$certURL=(Get-AzKeyVaultSecret -VaultName $keyvaultName -Name "mycert").id

$vml=Get-AzVM -ResourceGroupName $resourceGroup -Name "myVM"
$vaultId=(Get-AzKeyVault -ResourceGroupName $resourceGroup -VaultName $keyVaultName).ResourceId
$vm = Add-AzVMSecret -VM $vm -SourceVaultId $vaultId -CertificateStore "My" -CertificateUrl $certURL

Update-AzVM -ResourceGroupName $resourceGroup -VM $vm

```

## Configure IIS to use the certificate

Use the Custom Script Extension again with [Set-AzVMExtension](#) to update the IIS configuration. This update applies the certificate injected from Key Vault to IIS and configures the web binding:

```

$publicSettings = '{
    "fileUris": ["https://raw.githubusercontent.com/Azure-Samples/compute-automation-
configurations/master/secure-iis.ps1"],
    "commandToExecute": "powershell -ExecutionPolicy Unrestricted -File secure-iis.ps1"
}'

Set-AzVMExtension -ResourceGroupName $resourceGroup `-
    -ExtensionName "IIS" `-
    -VMName "myVM" `-
    -Location $location `-
    -Publisher "Microsoft.Compute" `-
    -ExtensionType "CustomScriptExtension" `-
    -TypeHandlerVersion 1.8 `-
    -SettingString $publicSettings

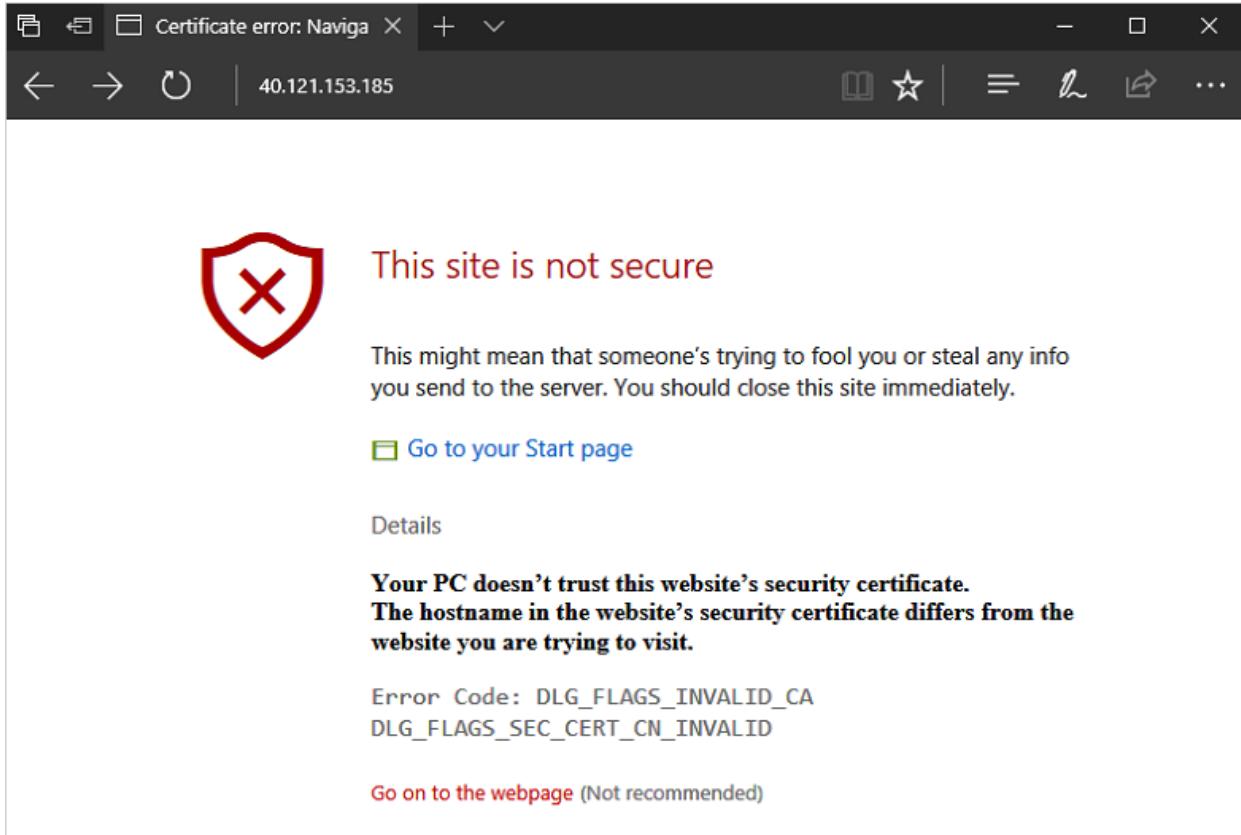
```

## Test the secure web app

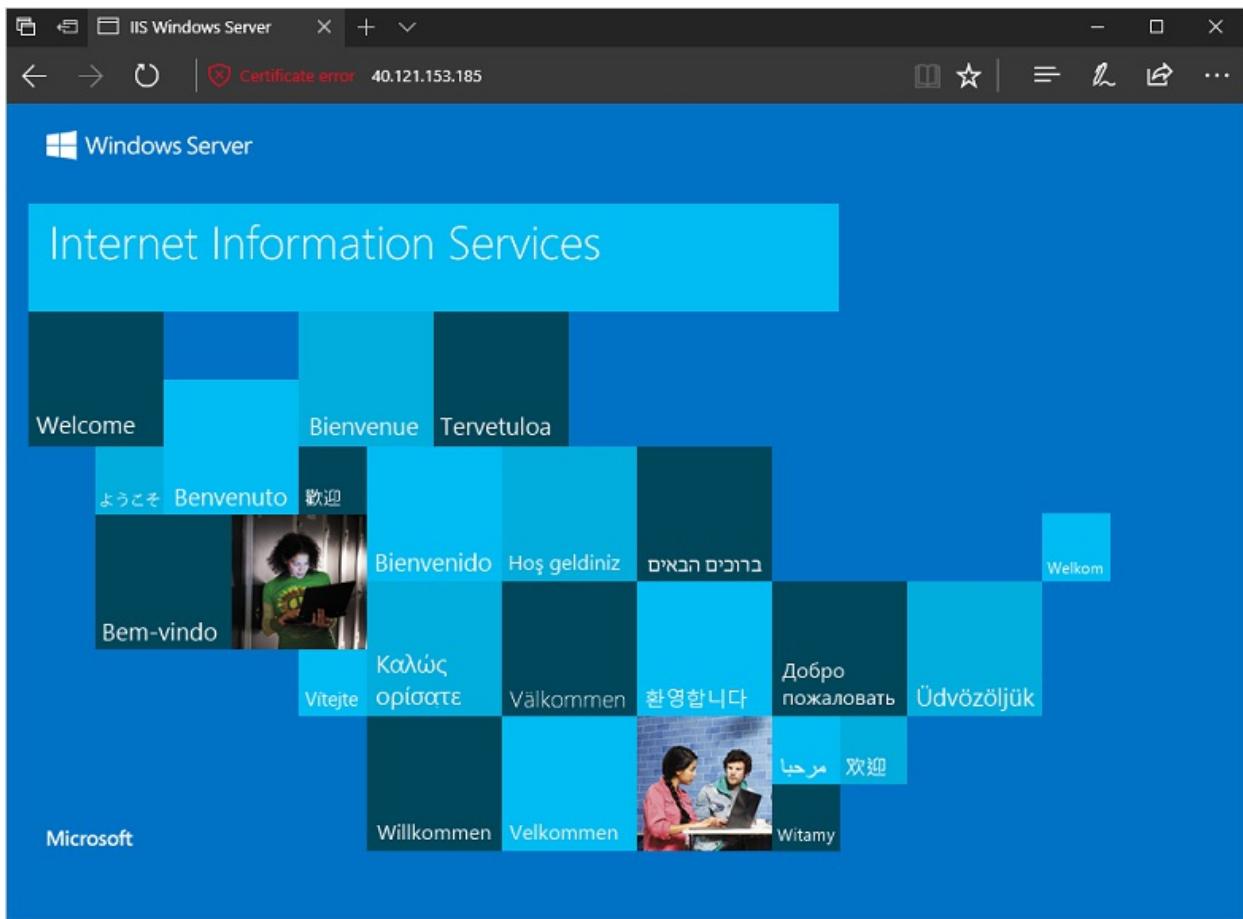
Obtain the public IP address of your VM with `Get-AzPublicIPAddress`. The following example obtains the IP address for `myPublicIP` created earlier:

```
Get-AzPublicIPAddress -ResourceGroupName $resourceGroup -Name "myPublicIPAddress" | select "IpAddress"
```

Now you can open a web browser and enter `https://<myPublicIP>` in the address bar. To accept the security warning if you used a self-signed certificate, select **Details** and then **Go on to the webpage**:



Your secured IIS website is then displayed as in the following example:



## Next steps

In this tutorial, you secured an IIS web server with a TLS/SSL certificate stored in Azure Key Vault. You learned how to:

- Create an Azure Key Vault
- Generate or upload a certificate to the Key Vault
- Create a VM and install the IIS web server
- Inject the certificate into the VM and configure IIS with a TLS binding

Follow this link to see pre-built virtual machine script samples.

[Windows virtual machine script samples](#)

# Delete a VM and attached resources

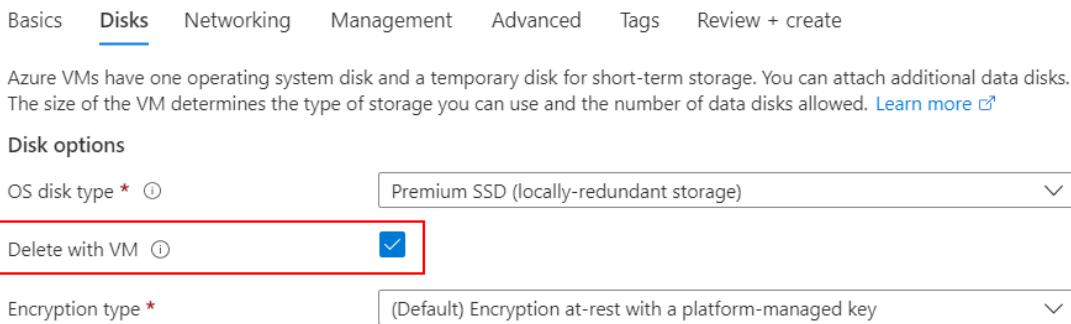
9/21/2022 • 8 minutes to read • [Edit Online](#)

Depending on how you delete a VM, it may only delete the VM resource, not the networking and disk resources. You can change the default settings for what other resources are deleted when you delete a VM.

## Set delete options when creating a VM

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

1. Open the [portal](#).
2. Select + **Create a resource**.
3. On the **Create a resource** page, under **Virtual machines**, select **Create**.
4. Make your choices on the **Basics**, then select **Next : Disks >**. The **Disks** tab will open.
5. Under **Disk options**, by default the OS disk is set to **Delete with VM**. If you don't want to delete the OS disk, clear the checkbox. If you're using an existing OS disk, the default is to detach the OS disk when the VM is deleted.



6. Under **Data disks**, you can either attach an existing data disk or create a new disk and attach it to the VM.
  - If you choose **Create and attach a new disk**, the **Create a new disk** page will open and you can select whether to delete the disk when you delete the VM.

## Create a new disk

Create a new disk to store applications and data on your VM. Disk pricing varies based on factors including disk size, storage type, and number of transactions. [Learn more](#)

Name *	<input type="text" value="deletetest_DataDisk_0"/>
Source type * ⓘ	<input type="text" value="None (empty disk)"/>
Size * ⓘ	<p>1024 GiB Premium SSD LRS <a href="#">Change size</a></p>
Encryption type *	<input type="text" value="(Default) Encryption at-rest with a platform-managed key"/>
Enable shared disk	<input type="radio"/> Yes <input checked="" type="radio"/> No
<input checked="" type="checkbox"/> Delete disk with VM <input type="checkbox"/>	

- If you choose to **Attach an existing disk**, you'll be able to choose the disk, LUN, and whether you want to delete the data disk when you delete the VM.

LUN	Name	Size (GiB)	Disk type	Host caching	Delete with VM ⓘ
0	No existing disk... ⏺			Read-only ⏺	<input type="checkbox"/>

[Create and attach a new disk](#) [Attach an existing disk](#)

- When you're done adding your disk information, select **Next : Networking**. The **Networking** tab will open.
- Towards the bottom of the page, select **Delete public IP and NIC when VM is deleted**.
- When you're done making selections, select **Review + create**. The **Review + create** page will open.
- You can verify which resources you have chosen to delete when you delete the VM.
- When you're satisfied with your selections, and validation passes, select **Create** to deploy the VM.

## Update the delete behavior on an existing VM

You can change the behavior when you delete a VM. The following example updates the VM to delete the NIC, OS disk, and data disk when the VM is deleted.

- [CLI](#)
- [REST](#)

The following example sets the delete option to `detach` so you can reuse the disk.

```
az resource update --resource-group myResourceGroup --name myVM --resource-type virtualMachines --namespace Microsoft.Compute --set properties.storageProfile.osDisk.deleteOption=detach
```

## Force Delete for VMs

Force delete allows you to forcefully delete your virtual machine, reducing delete latency and immediately freeing up attached resources. For VMs that do not require graceful shutdown, Force Delete will delete the VM as fast as possible while relieving the logical resources from the VM, bypassing the graceful shutdown and some of the cleanup operations. Force Delete will not immediately free the MAC address associated with a VM, as this is a physical resource that may take up to 10 min to free. If you need to immediately re-use the MAC address on a new VM, Force Delete is not recommended. Force delete should only be used when you are not intending to re-use virtual hard disks. You can use force delete through Portal, CLI, PowerShell, and REST API.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

When you go to delete an existing VM, you will find an option to apply force delete in the delete pane.

1. Open the [portal](#).
2. Navigate to your virtual machine.
3. On the **Overview** page, select **Delete**.
4. In the **Delete virtual machine** pane, select the checkbox for **Apply force delete**.
5. Select **Ok**.

## Force Delete for virtual machine scale sets

Force delete allows you to forcefully delete your **Uniform** virtual machine scale sets, reducing delete latency and immediately freeing up attached resources. Force Delete will not immediately free the MAC address associated with a VM, as this is a physical resource that may take up to 10 min to free. If you need to immediately re-use the MAC address on a new VM, Force Delete is not recommended. Force delete should only be used when you are not intending to re-use virtual hard disks. You can use force delete through Portal, CLI, PowerShell, and REST API.

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [REST](#)

When you go to delete an existing virtual machine scale set, you will find an option to apply force delete in the delete pane.

1. Open the [portal](#).
2. Navigate to your virtual machine scale set.
3. On the **Overview** page, select **Delete**.
4. In the **Delete virtual machine scale set** pane, select the checkbox for **Apply force delete**.
5. Select **Ok**.

## FAQ

### **Q: Does this feature work with shared disks?**

A: For shared disks, you can't set the 'deleteOption' property to 'Delete'. You can leave it blank or set it to 'Detach'

### **Q: Which Azure resources support this feature?**

A: This feature is supported on all managed disk types used as OS disks and Data disks, NICs, and Public IPs

**Q: Can I use this feature on disks and NICs that aren't associated with a VM?**

A: No, this feature is only available on disks and NICs associated with a VM.

**Q: How does this feature work with Flexible virtual machine scale sets?**

A: For Flexible virtual machine scale sets the disks, NICs, and PublicIPs have `deleteOption` set to `Delete` by default so these resources are automatically cleaned up when the VMs are deleted.

For data disks that were explicitly created and attached to the VMs, you can modify this property to 'Detach' instead of 'Delete' if you want the disks to persist after the VM is deleted.

**Q: Do Spot VMs support this feature?**

A: Yes, you can use this feature for Spot VMs just the way you would for on-demand VMs.

**Q: How do I persist the disks, NIC, and Public IPs associated with a VM?**

A: By default, disks, NICs, and Public IPs associated with a VM are persisted when the VM is deleted. If you configure these resources to be automatically deleted, you can update the settings so that the resources remain after the VM is deleted. To keep these resources, set the `deleteOption` property to `Detach`.

## Next steps

To learn more about basic VM management, see [Tutorial: Create and Manage Linux VMs with the Azure CLI](#).

# Connect to a Linux VM

9/21/2022 • 7 minutes to read • [Edit Online](#)

In Azure there are multiple ways to connect to a Linux virtual machine. The most common practice for connecting to a Linux VM is using the Secure Shell Protocol (SSH). This is done via any standard SSH client commonly found in Linux and Windows. You can also use [Azure Cloud Shell](#) from any browser.

This document describes how to connect, via SSH, to a VM that has a public IP. If you need to connect to a VM without a public IP, see [Azure Bastion Service](#).

## Prerequisites

- You need an SSH key pair. If you don't already have one, Azure will create a key pair during the deployment process. If you need help with creating one manually, see [Create and use an SSH public-private key pair for Linux VMs in Azure](#).
- You need an existing Network Security Group (NSG). Most VMs will have an NSG by default, but if you don't already have one you can create one and attach it manually. For more information, see [Create, change, or delete a network security group](#).
- To connect to a Linux VM, you need the appropriate port open. Typically this will be port 22. The following instructions assume port 22 but the process is the same for other port numbers. You can validate an appropriate port is open for SSH using the troubleshooter or by checking manually in your VM settings.

To check if port 22 is open:

1. On the page for the VM, select **Networking** from the left menu.
2. On the **Networking** page, check to see if there is a rule which allows TCP on port 22 from the IP address of the computer you are using to connect to the VM. If the rule exists, you can move to the next section.

The screenshot shows the Azure portal interface for a Linux VM named "LinuxVM". The left sidebar shows options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking (which is selected), Connect, and Disks. The main content area is titled "LinuxVM | Networking" and shows the "Virtual machine" blade. At the top, there are buttons for Attach network interface, Detach network interface, and Feedback. Below that, tabs for Inbound port rules, Outbound port rules, Application security groups, and Load balancing are visible. Under the Inbound port rules tab, it says "Network security group LinuxVM-nsg (attached to network interface: linuxvm395) Impacts 0 subnets, 1 network interfaces". There is a button labeled "Add inbound port rule". A table lists the current rules:

Priority	Name	Port	Protocol	Source
300	SSH	22	TCP	Any
65000	AllowVnetInBound	Any	Any	VirtualN
65001	AllowAzureLoadBala...	Any	Any	AzureLo
65500	DenyAllInBound	Any	Any	Any

3. If there isn't a rule, add one by selecting **Add inbound port rule**.
4. For **Service**, select **SSH** from the dropdown.

**Add inbound security rule**

Source ①

Any

Source port ranges \* ①

\*

Destination ①

Any

Service ①

SSH

Destination port ranges ①

22

Protocol

Any

TCP

UDP

ICMP

Action

Allow

Deny

Priority \* ①

100

Name \*

SSH

Description

Add Cancel

5. Edit Priority and Source if necessary

6. For Name, type *SSH*.

7. When you're done, select Add.

8. You should now have an SSH rule in the table of inbound port rules.

- Your VM must have a public IP address. To check if your VM has a public IP address, select **Overview** from the left menu and look at the **Networking** section. If you see an IP address next to **Public IP address**, then your VM has a public IP

If your VM does not have a public IP Address, it will look like this:

Networking	
Public IP address	-
Public IP address (IPv6)	-
Private IP address	10.1.0.4
Private IP address (IPv6)	-
Virtual network/subnet	myResourceGroup-vnet/default
DNS name	-

To learn more about adding a public IP address to an existing VM, see [Associate a public IP address to a virtual machine](#)

- Verify your VM is running. On the Overview tab, in the **Essentials** section, verify the status of the VM is **Running**. To start the VM, select **Start** at the top of the page.

## ^ Essentials

Resource group ([move](#))

LINUXVM

Status

Running

## Connect to the VM

Once the above prerequisites are met, you are ready to connect to your VM. Open your SSH client of choice. The SSH client command is typically included in Linux, macOS, and Windows. If you are using Windows 7 or older, where Win32 OpenSSH is not included by default, consider installing [WSL](#) or using [Azure Cloud Shell](#) from the browser.

### NOTE

The following examples assume the SSH key is in the key.pem format. If you used CLI or Azure PowerShell to download your keys, they may be in the id\_rsa format.

- [WSL, macOS, or native Linux client](#)
- [Windows command line \(cmd.exe, PowerShell etc.\)](#)

### SSH with a new key pair

- Ensure your public and private keys are in the correct directory. The directory is usually `~/.ssh`.

If you generated keys manually or generated them with the CLI, then the keys are probably already there. However, if you downloaded them in pem format from the Azure portal, you may need to move them to the right location. This can be done with the following syntax:

```
mv PRIVATE_KEY_SOURCE PRIVATE_KEY_DESTINATION
```

For example, if the key is in the `Downloads` folder, and `myKey.pem` is the name of your SSH key, type:

```
mv /Downloads/myKey.pem ~/.ssh
```

### NOTE

If you're using WSL, local files are found in the `/mnt/c/` directory. Accordingly, the path to the downloads folder and SSH key would be `/mnt/c/Users/{USERNAME}/Downloads/myKey.pem`

- Ensure you have read-only access to the private key by running

```
chmod 400 ~/.ssh/myKey.pem
```

- Run the SSH command with the following syntax: `ssh -i PATH_TO_PRIVATE_KEY USERNAME@EXTERNAL_IP`

For example, if your `azureuser` is the username you created and `20.51.230.13` is the public IP address of your VM, type:

```
ssh -i ~/.ssh/myKey.pem azureuser@20.51.230.13
```

#### 4. Validate the returned fingerprint.

If you have never connected to this VM before, you'll be asked to verify the hosts fingerprint. It's tempting to simply accept the fingerprint presented, but that exposes you to a potential person in the middle attack. You should always validate the hosts fingerprint. You only need to do this the first time you connect from a client. To get the host fingerprint via the portal, use the Run Command feature to execute the command:

```
ssh-keygen -lf /etc/ssh/ssh_host_ecdsa_key.pub | awk '{print $2}'
```

#### 5. Success! You should now be connected to your VM. If you're unable to connect, see [Troubleshoot SSH connections](#).

### SSH With existing public key

1. Run the following command in your SSH client. In this example, *20.51.230.13* is the public IP Address of your VM and *azureuser* is the username you created when you created the VM.

```
ssh azureuser@20.51.230.13
```

#### 2. Validate the returned fingerprint.

If you have never connected to this VM before you will be asked to verify the hosts fingerprint. It is tempting to simply accept the fingerprint presented, however, this exposes you to a possible person in the middle attack. You should always validate the hosts fingerprint. You only need to do this on the first time you connect from a client. To obtain the host fingerprint via the portal, use the Run Command feature to execute the command:

```
ssh-keygen -lf /etc/ssh/ssh_host_ecdsa_key.pub | awk '{print $2}'
```

#### 3. Success! You should now be connected to your VM. If you're unable to connect, see our troubleshooting guide [Troubleshoot SSH connections](#).

### Password authentication

#### WARNING

This type of authentication method is not as secure and is not recommended.

1. Run the following command in your SSH client. In this example, *20.51.230.13* is the public IP Address of your VM and *azureuser* is the username you created when you created the VM.

```
ssh azureuser@20.51.230.13
```

If you forgot your password or username see [Reset Access to an Azure VM](#)

#### 2. Validate the returned fingerprint.

If you have never connected to this VM before you will be asked to verify the hosts fingerprint. It is tempting to simply accept the fingerprint presented, however, this exposes you to a possible person in the middle attack. You should always validate the hosts fingerprint. You only need to do this on the first time

you connect from a client. To obtain the host fingerprint via the portal, use the Run Command feature to execute the command:

```
ssh-keygen -lf /etc/ssh/ssh_host_ecdsa_key.pub | awk '{print $2}'
```

3. Success! You should now be connected to your VM. If you're unable to connect using the correct method above, see [Troubleshoot SSH connections](#).

## Next steps

Learn how to transfer files to an existing VM, see [Use SCP to move files to and from a VM](#).

# Detailed steps: Create and manage SSH keys for authentication to a Linux VM in Azure

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

With a secure shell (SSH) key pair, you can create a Linux virtual machine that uses SSH keys for authentication. This article shows you how to create and use an SSH RSA public-private key file pair for SSH client connections.

If you want quick commands rather than a more in-depth explanation of SSH keys, see [How to create an SSH public-private key pair for Linux VMs in Azure](#).

To create SSH keys and use them to connect to a Linux VM from a **Windows** computer, see [How to use SSH keys with Windows on Azure](#). You can also use the [Azure portal](#) to create and manage SSH keys for creating VMs in the portal.

## Overview of SSH and keys

SSH is an encrypted connection protocol that provides secure sign-ins over unsecured connections. Although SSH provides an encrypted connection, using passwords with SSH connections still leaves the VM vulnerable to brute-force attacks. We recommend connecting to a VM over SSH using a public-private key pair, also known as *SSH keys*.

- The *public key* is placed on your VM.
- The *private key* remains on your local system. Protect this private key. Do not share it.

When you use an SSH client to connect to your VM (which has the public key), the remote VM tests the client to make sure it has the correct private key. If the client has the private key, it's granted access to the VM.

Depending on your organization's security policies, you can reuse a single public-private key pair to access multiple Azure VMs and services. You do not need a separate pair of keys for each VM or service you wish to access.

Your public key can be shared with anyone, but only you (or your local security infrastructure) should have access to your private key.

## Supported SSH key formats

Azure currently supports SSH protocol 2 (SSH-2) RSA public-private key pairs with a minimum length of 2048 bits. Other key formats such as ED25519 and ECDSA are not supported.

## SSH keys use and benefits

When you create an Azure VM by specifying the public key, Azure copies the public key (in the `.pub` format) to the `~/.ssh/authorized_keys` folder on the VM. SSH keys in `~/.ssh/authorized_keys` ensure that connecting clients present the corresponding private key during an SSH connection. In an Azure Linux VM that uses SSH keys for authentication, Azure disables the SSH server's password authentication system and only allows for SSH key authentication. By creating an Azure Linux VM with SSH keys, you can help secure the VM deployment and save yourself the typical post-deployment configuration step of disabling passwords in the `sshd_config` file.

If you do not wish to use SSH keys, you can set up your Linux VM to use password authentication. If your VM is

not exposed to the Internet, using passwords may be sufficient. However, you still need to manage your passwords for each Linux VM and maintain healthy password policies and practices, such as minimum password length and regular system updates.

## Generate keys with ssh-keygen

To create the keys, a preferred command is `ssh-keygen`, which is available with OpenSSH utilities in the Azure Cloud Shell, a macOS or Linux host, and Windows (10 & 11). `ssh-keygen` asks a series of questions and then writes a private key and a matching public key.

SSH keys are by default kept in the `~/.ssh` directory. If you do not have a `~/.ssh` directory, the `ssh-keygen` command creates it for you with the correct permissions. An SSH key is created as a resource and stored in Azure for later use.

### NOTE

You can also create keys with the [Azure CLI](#) with the `az sshkey create` command, as described in [Generate and store SSH keys](#).

### Basic example

The following `ssh-keygen` command generates 4096-bit SSH RSA public and private key files by default in the `~/.ssh` directory. If an existing SSH key pair is found in the current location, those files are overwritten.

```
ssh-keygen -m PEM -t rsa -b 4096
```

### Detailed example

The following example shows additional command options to create an SSH RSA key pair. If an SSH key pair exists in the current location, those files are overwritten.

```
ssh-keygen \
-m PEM \
-t rsa \
-b 4096 \
-C "azureuser@myserver" \
-f ~/.ssh/mykeys/myprivatekey \
-N mypassphrase
```

### Command explained

`ssh-keygen` = the program used to create the keys

`-m PEM` = format the key as PEM

`-t rsa` = type of key to create, in this case in the RSA format

`-b 4096` = the number of bits in the key, in this case 4096

`-C "azureuser@myserver"` = a comment appended to the end of the public key file to easily identify it. Normally an email address is used as the comment, but use whatever works best for your infrastructure.

`-f ~/.ssh/mykeys/myprivatekey` = the filename of the private key file, if you choose not to use the default name. A corresponding public key file appended with `.pub` is generated in the same directory. The directory must exist.

`-N mypassphrase` = an additional passphrase used to access the private key file.

## Example of ssh-keygen

```
ssh-keygen -t rsa -m PEM -b 4096 -C "azureuser@myserver"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/azureuser/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/azureuser/.ssh/id_rsa.
Your public key has been saved in /home/azureuser/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:vFfHHrpSGQD/oNdvNiX0sG9Vh+wR0lZBktNZw9AUjA azureuser@myserver
The key's randomart image is:
+---[RSA 4096]----+
|       .oE=*B*+ |
|       o+o.*++|
|       .oo++*|
|       . .B+.O|
|       S   o=BO.|
|       . .o++o |
|       . .... .|
|       .. . . |
|       .. . |
+---[SHA256]-----+
```

### Saved key files

Enter file in which to save the key (/home/azureuser/.ssh/id\_rsa): `~/.ssh/id_rsa`

The key pair name for this article. Having a key pair named `id_rsa` is the default; some tools might expect the `id_rsa` private key file name, so having one is a good idea. The directory `~/ssh/` is the default location for SSH key pairs and the SSH config file. If not specified with a full path, `ssh-keygen` creates the keys in the current working directory, not the default `~/ssh`.

### List of the `~/ssh` directory

To view existing files in the `~/ssh` directory, run the following command. If no files are found in the directory or the directory itself is missing, make sure that all previous commands were successfully run. You may require root access to modify files in this directory on certain Linux distributions.

```
ls -al ~/ssh
-rw----- 1 azureuser staff 1675 Aug 25 18:04 id_rsa
-rw-r--r-- 1 azureuser staff 410 Aug 25 18:04 id_rsa.pub
```

### Key passphrase

Enter passphrase (empty for no passphrase):

It is *strongly* recommended to add a passphrase to your private key. Without a passphrase to protect the key file, anyone with the file can use it to sign in to any server that has the corresponding public key. Adding a passphrase offers more protection in case someone is able to gain access to your private key file, giving you time to change the keys.

## Generate keys automatically during deployment

If you use the [Azure CLI](#) to create your VM, you can optionally generate both public and private SSH key files by running the `az vm create` command with the `--generate-ssh-keys` option. The keys are stored in the `~/ssh` directory. Note that this command option does not overwrite keys if they already exist in that location, such as with some pre-configured Compute Gallery images.

## Provide SSH public key when deploying a VM

To create a Linux VM that uses SSH keys for authentication, provide your SSH public key when creating the VM using the Azure portal, CLI, Resource Manager templates, or other methods. When using the portal, you enter the public key itself. If you use the [Azure CLI](#) to create your VM with an existing public key, specify the value or location of this public key by running the `az vm create` command with the `--ssh-key-value` option.

If you're not familiar with the format of an SSH public key, you can see your public key by running `cat` as follows, replacing `~/.ssh/id_rsa.pub` with your own public key file location:

```
cat ~/.ssh/id_rsa.pub
```

Output is similar to the following (redacted example below):

```
ssh-rsa  
XXXXXXXXXc2EAAAADAXABAAABAXC5Am7+fGZ+5zXBGgXS6GUvmsXCLGc7tX7/rViXk3+eShZzaXnt75gUmT1I2f75zFn2h1AIDGKWF4g12K  
WcZxy81TniUOTjUsVlwPymXUXxESL/UfJKfbdstBhT0dy5EG9rYWA0K43SJmwPhH28BpoLfXXXXXG+/ilsXXXXXgRLiJ2W19MzXHp8z3Lxw  
7r9wx3HaV1P4XiFv9U4h6cp8RMI1MP1nNesFl0BpG4pV2bJRBTXNxY416F8WZ3C4ku8Xx0o8mxaTpVZ3T1841altnMTZCcPkXuMrBjYSJ  
bA8npoXAXNwiivyo3X2KMXXXXdXXXXXXXXXXXX/ azureuser@myserver
```

If you copy and paste the contents of the public key file into the Azure portal or a Resource Manager template, make sure you don't copy any additional whitespace or introduce additional line breaks. For example, if you use macOS, you can pipe the public key file (by default, `~/.ssh/id_rsa.pub`) to `pbcopy` to copy the contents (there are other Linux programs that do the same thing, such as `xclip`).

If you prefer to use a public key that is in a multiline format, you can generate an RFC4716 formatted key in a 'pem' container from the public key you previously created.

To create a RFC4716 formatted key from an existing SSH public key:

```
ssh-keygen \  
-f ~/.ssh/id_rsa.pub \  
-e \  
-m RFC4716 > ~/.ssh/id_ssh2.pem
```

## SSH to your VM with an SSH client

With the public key deployed on your Azure VM, and the private key on your local system, SSH to your VM using the IP address or DNS name of your VM. Replace `azureuser` and `myvm.westus.cloudapp.azure.com` in the following command with the administrator user name and the fully qualified domain name (or IP address):

```
ssh azureuser@myvm.westus.cloudapp.azure.com
```

If you provided a passphrase when you created your key pair, enter the passphrase when prompted during the sign-in process. (The server is added to your `~/.ssh/known_hosts` folder, and you won't be asked to connect again until the public key on your Azure VM changes or the server name is removed from `~/.ssh/known_hosts`.)

If the VM is using the just-in-time access policy, you need to request access before you can connect to the VM. For more information about the just-in-time policy, see [Manage virtual machine access using the just in time policy](#).

## Use ssh-agent to store your private key passphrase

To avoid typing your private key file passphrase with every SSH sign-in, you can use `ssh-agent` to cache your private key file passphrase on your local system. If you are using a Mac, the macOS Keychain securely stores the

private key passphrase when you invoke `ssh-agent`.

Verify and use `ssh-agent` and `ssh-add` to inform the SSH system about the key files so that you do not need to use the passphrase interactively.

```
eval "$(ssh-agent -s)"
```

Now add the private key to `ssh-agent` using the command `ssh-add`.

```
ssh-add ~/.ssh/id_rsa
```

The private key passphrase is now stored in `ssh-agent`.

## Use `ssh-copy-id` to copy the key to an existing VM

If you have already created a VM, you can add a new SSH public key to your Linux VM using `ssh-copy-id`.

```
ssh-copy-id -i ~/.ssh/id_rsa.pub azureuser@myserver
```

## Create and configure an SSH config file

You can create and configure an SSH config file (`~/.ssh/config`) to speed up log-ins and to optimize your SSH client behavior.

The following example shows a simple configuration that you can use to quickly sign in as a user to a specific VM using the default SSH private key.

Create the file.

```
touch ~/.ssh/config
```

Edit the file to add the new SSH configuration

```
vim ~/.ssh/config
```

Add configuration settings appropriate for your host VM. In this example, the VM name (Host) is *myvm*, the account name (User) is *azureuser* and the IP Address or FQDN (Hostname) is 192.168.0.255.

```
# Azure Keys
Host myvm
  Hostname 192.168.0.255
  User azureuser
# ./Azure Keys
```

You can add configurations for additional hosts to enable each to use its own dedicated key pair. See [SSH config file](#) for more advanced configuration options.

Now that you have an SSH key pair and a configured SSH config file, you are able to remotely access your Linux VM quickly and securely. When you run the following command, SSH locates and loads any settings from the `Host myvm` block in the SSH config file.

```
ssh myvm
```

The first time you sign in to a server using an SSH key, the command prompts you for the passphrase for that key file.

## Next steps

Next up is to create Azure Linux VMs using the new SSH public key. Azure VMs that are created with an SSH public key as the sign-in are better secured than VMs created with the default sign-in method, passwords.

- [Create a Linux virtual machine with the Azure portal](#)
- [Create a Linux virtual machine with the Azure CLI](#)
- [Create a Linux VM using an Azure template](#)

# Generate and store SSH keys in the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

If you frequently use the portal to deploy Linux VMs, you can make using SSH keys simpler by creating them directly in the portal, or uploading them from your computer.

You can create a SSH keys when you first create a VM, and reuse them for other VMs. Or, you can create SSH keys separately, so that you have a set of keys stored in Azure to fit your organizations needs.

If you have existing keys and you want to simplify using them in the portal, you can upload them and store them in Azure for reuse.

For more detailed information about creating and using SSH keys with Linux VMs, see [Use SSH keys to connect to Linux VMs](#).

## Generate new keys

1. Open the [Azure portal](#).
2. At the top of the page, type *SSH* to search. Under **Marketplace**, select **SSH keys**.
3. On the **SSH Key** page, select **Create**.

[Home](#) > [SSH Key](#) >

### Create an SSH key

[Basics](#)   [Tags](#)   [Review + create](#)

Creating an SSH key resource allows you to manage and use public keys stored in Azure with Linux virtual machines.  
[Learn more](#)

#### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Pay-As-You-Go

Resource group \* ⓘ

(New) mySSHKeyGroup

[Create new](#)

#### Instance details

Region \* ⓘ

(US) East US

Key pair name \*

mySSHKey1

SSH public key source

Generate new key pair

[Review + create](#)

< Previous

Next : Tags >

4. In **Resource group** select **Create new** to create a new resource group to store your keys. Type a name

for your resource group and select **OK**.

5. In **Region** select a region to store your keys. You can use the keys in any region, this is just the region where they will be stored.
6. Type a name for your key in **Key pair name**.
7. In **SSH public key source**, select **Generate public key source**.
8. When you are done, select **Review + create**.
9. After it passes validation, select **Create**.
10. You will then get a pop-up window to, select **Download private key and create resource**. This will download the SSH key as a .pem file.

## Generate new key pair

**i** An SSH key pair contains both a public key and a private key. **Azure doesn't store the private key**. After the SSH key resource is created, you won't be able to download the private key again. [Learn more](#)

**Download private key and create resource**

[Return to create an SSH key resource](#)

11. Once the .pem file is downloaded, you might want to move it somewhere on your computer where it is easy to point to from your SSH client.

## Connect to the VM

On your local computer, open a PowerShell prompt and type:

```
ssh -i <path to the .pem file> username@<ipaddress of the VM>
```

For example, type: `ssh -i /Downloads/mySSHKey.pem azureuser@123.45.67.890`

## Upload an SSH key

You can also upload a public SSH key to store in Azure. For information about how to create an SSH key pair, see [Use SSH keys to connect to Linux VMs](#).

1. Open the [Azure portal](#).
2. At the top of the page, type *SSH* to search. Under \**Marketplace*, select **SSH keys**.
3. On the **SSH Key** page, select **Create**.

## Create an SSH key

Basics Tags Review + create

Creating an SSH key resource allows you to manage and use public keys stored in Azure with Linux virtual machines.  
[Learn more](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Pay-As-You-Go

Resource group \* ⓘ

(New) mySSHKeyGroup

[Create new](#)

### Instance details

Region \* ⓘ

(US) East US

Key pair name \*

mySSHKey1

SSH public key source

Upload existing public key

Upload key \* ⓘ

Paste the public key here to upload

[Learn more about creating and using SSH keys in Azure](#)

[Review + create](#)

< Previous

Next : Tags >

4. In **Resource group** select **Create new** to create a new resource group to store your keys. Type a name for your resource group and select **OK**.
5. In **Region** select a region to store your keys. You can use the keys in any region, this is just the region where they will be stored.
6. Type a name for your key in **Key pair name**.
7. In **SSH public key source**, select **Upload existing public key**.
8. Paste the full contents of the public key into **Upload key** and then select **Review + create**.
9. After validation completes, select **Create**.

Once the key has been uploaded, you can choose to use it when you create a VM.

## List keys

SSH keys created in the portal are stored as resources, so you can filter your resources view to see all of them.

1. In the portal, select **All resource**.
2. In the filters, select **Type**, unselect the **Select all** option to clear the list.
3. Type **SSH** in the filter and select **SSH key**.



## Get the public key

If you need your public key, you can easily copy it from the portal page for the key. Just list your keys (using the process in the last section) then select a key from the list. The page for your key will open and you can click the **Copy to clipboard** icon next to the key to copy it.

## Next steps

To learn more about using SSH keys with Azure VMs, see [Use SSH keys to connect to Linux VMs](#).

# Generate and store SSH keys with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can create SSH keys before creating a VM, and store them in Azure. Each newly created SSH key is also stored locally.

If you have existing SSH keys, you can upload and store them in Azure for reuse.

For a more detailed overview of SSH, see [Detailed steps: Create and manage SSH keys for authentication to a Linux VM in Azure](#).

For more detailed information about creating and using SSH keys with Linux VMs, see [Use SSH keys to connect to Linux VMs](#).

## Generate new keys

1. After you sign in, use the [az sshkey create](#) command to create the new SSH key:

```
az sshkey create --name "mySSHKey" --resource-group "myResourceGroup"
```

2. The resulting output lists the new key files' paths:

```
Private key is saved to "/home/user/.ssh/7777777777_9999999".
Public key is saved to "/home/user/.ssh/7777777777_9999999.pub".
```

3. Change the permissions for the private key file for privacy:

```
chmod 600 /home/user/.ssh/7777777777_9999999
```

## Connect to the VM

On your local computer, open a Bash prompt:

```
ssh -i <path to the private key file> username@<ipaddress of the VM>
```

For example, enter: `ssh -i /home/user/.ssh/mySSHKey azureuser@123.45.67.890`

## Upload an SSH key

You can upload a public SSH key to store in Azure.

Use the [az sshkey create](#) command to upload an SSH public key by specifying its file:

```
az sshkey create --name "mySSHKey" --public-key "@/home/user/.ssh/7777777777_9999999.pub" --resource-group "myResourceGroup"
```

## List keys

Use the [az sshkey list](#) command to list all public SSH keys, optionally specifying a resource group:

```
az sshkey list --resource-group "myResourceGroup"
```

## Get the public key

Use the [az sshkey show](#) command to show the values of a public SSH key:

```
az sshkey show --name "mySSHKey" --resource-group "myResourceGroup"
```

## Next steps

To learn more about using SSH keys with Azure VMs, see [Use SSH keys to connect to Linux VMs](#).

# Quick steps: Create and use an SSH public-private key pair for Linux VMs in Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

With a secure shell (SSH) key pair, you can create virtual machines (VMs) in Azure that use SSH keys for authentication. This article shows you how to quickly generate and use an SSH public-private key file pair for Linux VMs. You can complete these steps with the Azure Cloud Shell, a macOS, or a Linux host.

For help with troubleshooting issues with SSH, see [Troubleshoot SSH connections to an Azure Linux VM that fails, errors out, or is refused](#).

## NOTE

VMs created using SSH keys are by default configured with passwords disabled, which greatly increases the difficulty of brute-force guessing attacks.

For more background and examples, see [Detailed steps to create SSH key pairs](#).

For additional ways to generate and use SSH keys on a Windows computer, see [How to use SSH keys with Windows on Azure](#).

## Supported SSH key formats

Azure currently supports SSH protocol 2 (SSH-2) RSA public-private key pairs with a minimum length of 2048 bits. Other key formats such as ED25519 and ECDSA are not supported.

## Create an SSH key pair

Use the `ssh-keygen` command to generate SSH public and private key files. By default, these files are created in the `~/.ssh` directory. You can specify a different location, and an optional password (*passphrase*) to access the private key file. If an SSH key pair with the same name exists in the given location, those files are overwritten.

The following command creates an SSH key pair using RSA encryption and a bit length of 4096:

```
ssh-keygen -m PEM -t rsa -b 4096
```

## NOTE

You can also create key pairs with the [Azure CLI](#) with the `az sshkey create` command, as described in [Generate and store SSH keys](#).

If you use the [Azure CLI](#) to create your VM with the `az vm create` command, you can optionally generate SSH public and private key files using the `--generate-ssh-keys` option. The key files are stored in the `~/.ssh` directory unless specified otherwise with the `--ssh-dest-key-path` option. If an ssh key pair already exists and the `--generate-ssh-keys` option is used, a new key pair will not be generated but instead the existing key pair will be used. In the following command, replace *VMname* and *RGname* with your own values:

```
az vm create --name VMname --resource-group RGname --image UbuntuLTS --generate-ssh-keys
```

## Provide an SSH public key when deploying a VM

To create a Linux VM that uses SSH keys for authentication, specify your SSH public key when creating the VM using the Azure portal, Azure CLI, Azure Resource Manager templates, or other methods:

- [Create a Linux virtual machine with the Azure portal](#)
- [Create a Linux virtual machine with the Azure CLI](#)
- [Create a Linux VM using an Azure template](#)

If you're not familiar with the format of an SSH public key, you can display your public key with the following

`cat` command, replacing `~/.ssh/id_rsa.pub` with the path and filename of your own public key file if needed:

```
cat ~/.ssh/id_rsa.pub
```

A typical public key value looks like this example:

```
ssh-rsa
AAAAB3NzaC1yc2EAAQABADQABAAQACQ1/KanayNr+Q7ogR5mKnGpKWRBQU7F3JjhN7utdf7Z2iUFykaYx+MInSnT3XdnBRS8KhC0IP8ptbng
IaNOWd6zM8hB6UrcRT1Tpwk/SuGMw1Vb40x1EFphBkVEUgBolOoANIEXriAMv1DMZsgvnMFiQ12tD/u14cxy1WNEMAftey/vX3Fgp2vEq4zH
XE1iY/sFZLJUJzcRUI0MOFHXAuCjg/qyqqbIuTDFyfg8k0JTtyGFEMQhbXKcuP2yGx1uw0ice62LRzr8w0mszftXyMik1PnshRXbmE2xgINY
g5xo/ra3mq2imwtOKJpfdtFoMiKhJmSNHBSk7vFTeYgg0v2cQ2+vL381cIFX40h+QCzvNF/Ax0DV1QtVtSqfQxRVG79Zqio5p12gHFkt1fV
7recBvVIhyx2L1YUkrq4DHzkxNY5c90GSHX5le9Ys03F1J5ip18f6gPq4xFmo6dVoJodZm9N0YMKCkZ4k1qJDE5sJBk2ujDPmQqeMjJX3Fn
DXYYB182ZCGQzXfz1PDC29cWvgDZEXNHuYr0LmJTmYtLZ4WkdUhLLlt5XsdoKwqlWpbegyYtGzeZNrt00dN6ybOPJqmYFd2qRtb4sYPniGJ
DOGhx4VodXAjT09omhQJpE6w1ZbRWdVkc55R2d/CSPHJscEiuudb+1SG2uA/oik/WQ== username@domainname
```

If you copy and paste the contents of the public key file to use in the Azure portal or a Resource Manager template, make sure you don't copy any trailing whitespace. To copy a public key in macOS, you can pipe the public key file to `pbcopy`. Similarly in Linux, you can pipe the public key file to programs such as `xclip`.

The public key that you place on your Linux VM in Azure is by default stored in `~/.ssh/id_rsa.pub`, unless you specified a different location when you created the key pair. To use the [Azure CLI 2.0](#) to create your VM with an existing public key, specify the value and optionally the location of this public key using the [az vm create](#) command with the `--ssh-key-values` option. In the following command, replace *myVM*, *myResourceGroup*, *UbuntuLTS*, *azureuser*, and *mysshkey.pub* with your own values:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image UbuntuLTS \
--admin-username azureuser \
--ssh-key-values mysshkey.pub
```

If you want to use multiple SSH keys with your VM, you can enter them in a space-separated list, like this

`--ssh-key-values sshkey-desktop.pub sshkey-laptop.pub`.

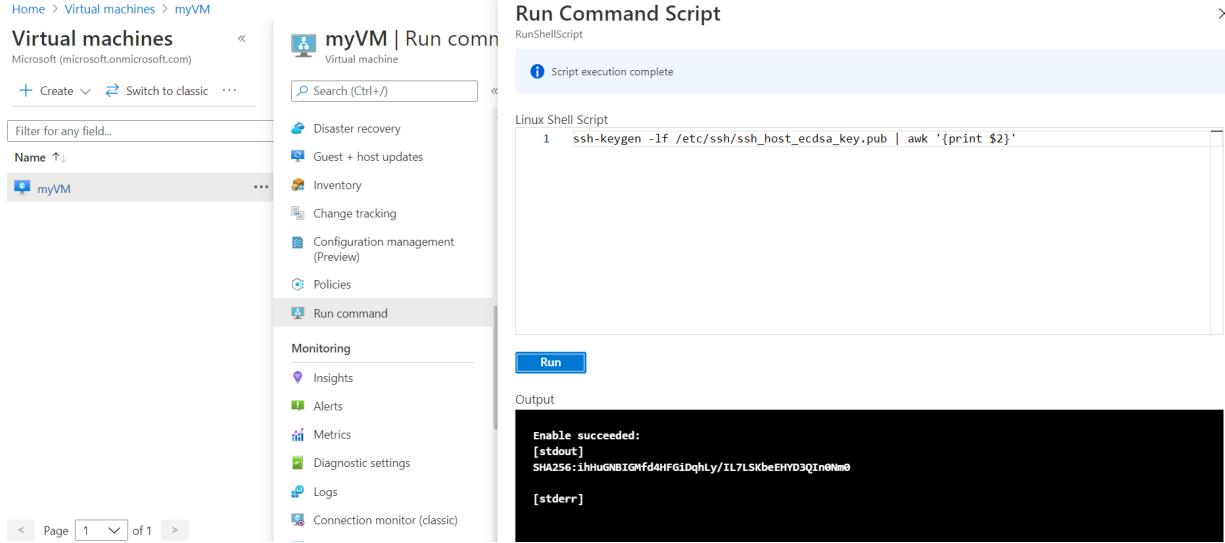
## SSH into your VM

With the public key deployed on your Azure VM, and the private key on your local system, SSH into your VM using the IP address or DNS name of your VM. In the following command, replace *azureuser* and *myvm.westus.cloudapp.azure.com* with the administrator user name and the fully qualified domain name (or IP address):

```
ssh azureuser@myvm.westus.cloudapp.azure.com
```

If you're connecting to this VM for the first time, you'll be asked to verify the host's fingerprint. It's tempting to simply accept the fingerprint that's presented, but that approach exposes you to a possible person-in-the-middle attack. You should always validate the host's fingerprint. You need to do this only the first time you connect from a client. To obtain the host fingerprint via the portal, use the Run Command feature to execute the command

```
ssh-keygen -lf /etc/ssh/ssh_host_ecdsa_key.pub | awk '{print $2}' .
```



To run the command using CLI, use [az vm run-command invoke](#).

If you specified a passphrase when you created your key pair, enter that passphrase when prompted during the sign-in process. The VM is added to your `~/.ssh/known_hosts` file, and you won't be asked to connect again until either the public key on your Azure VM changes or the server name is removed from `~/.ssh/known_hosts`.

If the VM is using the just-in-time access policy, you need to request access before you can connect to the VM. For more information about the just-in-time policy, see [Manage virtual machine access using the just in time policy](#).

## Next steps

- For more information on working with SSH key pairs, see [Detailed steps to create and manage SSH key pairs](#).
- If you have difficulties with SSH connections to Azure VMs, see [Troubleshoot SSH connections to an Azure Linux VM](#).

# How to use SSH keys with Windows on Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article is for Windows users who want to [create](#) and use *secure shell* (SSH) keys to [connect](#) to Linux virtual machines (VMs) in Azure. You can also [generate and store SSH keys in the Azure portal](#) to use when creating VMs in the portal.

To use SSH keys from a Linux or macOS client, see the [quick steps](#). For a more detailed overview of SSH, see [Detailed steps: Create and manage SSH keys for authentication to a Linux VM in Azure](#).

## Overview of SSH and keys

SSH is an encrypted connection protocol that allows secure sign-ins over unsecured connections. SSH is the default connection protocol for Linux VMs hosted in Azure. Although SSH itself provides an encrypted connection, using passwords with SSH still leaves the VM vulnerable to brute-force attacks. We recommend connecting to a VM over SSH using a public-private key pair, also known as *SSH keys*.

The public-private key pair is like the lock on your front door. The lock is exposed to the **public**, anyone with the right key can open the door. The key is **private**, and only given to people you trust because it can be used to unlock the door.

- The *public key* is placed on your Linux VM when you create the VM.
- The *private key* remains on your local system. Protect this private key. Do not share it.

When you connect to your Linux VM, the VM tests the SSH client to make sure it has the correct private key. If the client has the private key, it's granted access to the VM.

Depending on your organization's security policies, you can reuse a single key pair to access multiple Azure VMs and services. You do not need a separate pair of keys for each VM.

Your public key can be shared with anyone, but only you (or your local security infrastructure) should have access to your private key.

## Supported SSH key formats

Azure currently supports SSH protocol 2 (SSH-2) RSA public-private key pairs with a minimum length of 2048 bits. Other key formats such as ED25519 and ECDSA are not supported.

## SSH clients

Recent versions of Windows 10 include [OpenSSH client commands](#) to create and use SSH keys and make SSH connections from PowerShell or a command prompt.

You can also use Bash in the [Azure Cloud Shell](#) to connect to your VM. You can use Cloud Shell in a [web browser](#), from the [Azure portal](#), or as a terminal in Visual Studio Code using the [Azure Account extension](#).

You can also install the [Windows Subsystem for Linux](#) to connect to your VM over SSH and use other native Linux tools within a Bash shell.

## Create an SSH key pair

The easiest way to create and manage your SSH keys is to [use the portal to create and store them for reuse](#).

You can also create key pairs with the [Azure CLI](#) with the `az sshkey create` command, as described in [Generate and store SSH keys](#).

To create an SSH key pair on your local computer using the `ssh-keygen` command from PowerShell or a command prompt, type the following:

```
ssh-keygen -m PEM -t rsa -b 2048
```

Enter a filename, or use the default shown in parenthesis (for example `C:\Users\username/.ssh/id_rsa`). Enter a passphrase for the file, or leave the passphrase blank if you do not want to use a passphrase.

## Create a VM using your key

To create a Linux VM that uses SSH keys for authentication, provide your SSH public key when creating the VM.

Using the Azure CLI, you specify the path and filename for the public key using `az vm create` and the `--ssh-key-value` parameter.

```
az vm create \
  --resource-group myResourceGroup \
  --name myVM \
  --image UbuntuLTS \
  --admin-username azureuser \
  --ssh-key-value ~/.ssh/id_rsa.pub
```

With PowerShell, use `New-AzVM` and add the SSH key to the VM configuration using```. For an example, see [Quickstart: Create a Linux virtual machine in Azure with PowerShell](#).

If you do a lot of deployments using the portal, you might want to upload your public key to Azure, where it can be easily selected when creating a VM from the portal. For more information, see [Upload an SSH key](#).

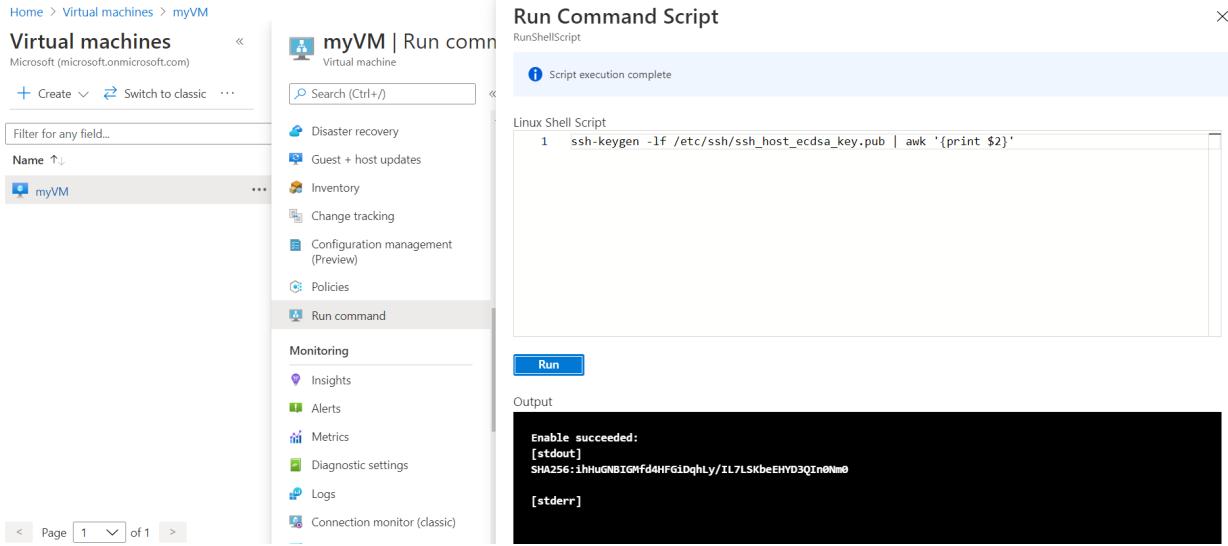
## Connect to your VM

With the public key deployed on your Azure VM, and the private key on your local system, SSH to your VM using the IP address or DNS name of your VM. Replace `azureuser` and `10.111.12.123` in the following command with the administrator user name, the IP address (or fully qualified domain name), and the path to your private key:

```
ssh -i ~/.ssh/id_rsa azureuser@10.111.12.123
```

If you have never connected to this VM before you will be asked to verify the hosts fingerprint. It is tempting to simply accept the fingerprint presented, however, this exposes you to a possible person in the middle attack. You should always validate the hosts fingerprint. You only need to do this on the first time you connect from a client. To obtain the host fingerprint via the portal use the Run Command with the following:

```
ssh-keygen -lf /etc/ssh/ssh_host_ecdsa_key.pub | awk '{print $2}' .
```



To run the command using CLI, use the `az vm run-command invoke` command.

If you configured a passphrase when you created your key pair, enter the passphrase when prompted.

If the VM is using the just-in-time access policy, you need to request access before you can connect to the VM.

For more information about the just-in-time policy, see [Manage virtual machine access using the just in time policy](#).

## Next steps

- For information about SSH keys in the Azure portal to use when creating VMs, see [Generate and store SSH keys in the Azure portal](#).
- For information about SSH keys in the Azure CLI to use when creating VMs, see [Generate and store SSH keys with the Azure CLI](#).
- For detailed steps, options, and advanced examples of working with SSH keys, see [Detailed steps to create SSH key pairs](#).
- You can also use PowerShell in Azure Cloud Shell to generate SSH keys and make SSH connections to Linux VMs. See the [PowerShell quickstart](#).
- If you have difficulty using SSH to connect to your Linux VMs, see [Troubleshoot SSH connections to an Azure Linux VM](#).

# Install and configure xrdp to use Remote Desktop with Ubuntu

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Linux virtual machines (VMs) in Azure are usually managed from the command line using a secure shell (SSH) connection. When new to Linux, or for quick troubleshooting scenarios, the use of remote desktop may be easier. This article details how to install and configure a desktop environment ([xfce](#)) and remote desktop ([xrdp](#)) for your Linux VM running Ubuntu.

The article was written and tested using an Ubuntu 18.04 VM.

## Prerequisites

This article requires an existing Ubuntu 18.04 LTS VM in Azure. If you need to create a VM, use one of the following methods:

- The [Azure CLI](#)
- The [Azure portal](#)

## Install a desktop environment on your Linux VM

Most Linux VMs in Azure do not have a desktop environment installed by default. Linux VMs are commonly managed using SSH connections rather than a desktop environment. There are various desktop environments in Linux that you can choose. Depending on your choice of desktop environment, it may consume one to 2 GB of disk space, and take 5 to 10 minutes to install and configure all the required packages.

The following example installs the lightweight [xfce4](#) desktop environment on an Ubuntu 18.04 LTS VM.

Commands for other distributions vary slightly (use `yum` to install on Red Hat Enterprise Linux and configure appropriate `selinux` rules, or use `zypper` to install on SUSE, for example).

First, SSH to your VM. The following example connects to the VM named `myvm.westus.cloudapp.azure.com` with the username of `azureuser`. Use your own values:

```
ssh azureuser@myvm.westus.cloudapp.azure.com
```

If you are using Windows and need more information on using SSH, see [How to use SSH keys with Windows](#).

Next, install xfce using `apt` as follows:

```
sudo apt-get update
sudo DEBIAN_FRONTEND=noninteractive apt-get -y install xfce4
sudo apt install xfce4-session
```

## Install and configure a remote desktop server

Now that you have a desktop environment installed, configure a remote desktop service to listen for incoming connections. [xrdp](#) is an open source Remote Desktop Protocol (RDP) server that is available on most Linux distributions, and works well with xfce. Install xrdp on your Ubuntu VM as follows:

```
sudo apt-get -y install xrdp  
sudo systemctl enable xrdp
```

On Ubuntu 20, you'll need to give certificate access to an xrdp user:

```
sudo adduser xrdp ssl-cert
```

Tell xrdp what desktop environment to use when you start your session. Configure xrdp to use xfce as your desktop environment as follows:

```
echo xfce4-session >~/.xsession
```

Restart the xrdp service for the changes to take effect as follows:

```
sudo service xrdp restart
```

## Set a local user account password

If you created a password for your user account when you created your VM, skip this step. If you only use SSH key authentication and do not have a local account password set, specify a password before you use xrdp to log in to your VM. xrdp cannot accept SSH keys for authentication. The following example specifies a password for the user account *azureuser*.

```
sudo passwd azureuser
```

### NOTE

Specifying a password does not update your SSHD configuration to permit password logins if it currently does not. From a security perspective, you may wish to connect to your VM with an SSH tunnel using key-based authentication and then connect to xrdp. If so, skip the following step on creating a network security group rule to allow remote desktop traffic.

## Create a Network Security Group rule for Remote Desktop traffic

To allow Remote Desktop traffic to reach your Linux VM, a network security group rule needs to be created that allows TCP on port 3389 to reach your VM. For more information about network security group rules, see [What is a network security group?](#) You can also [use the Azure portal to create a network security group rule](#).

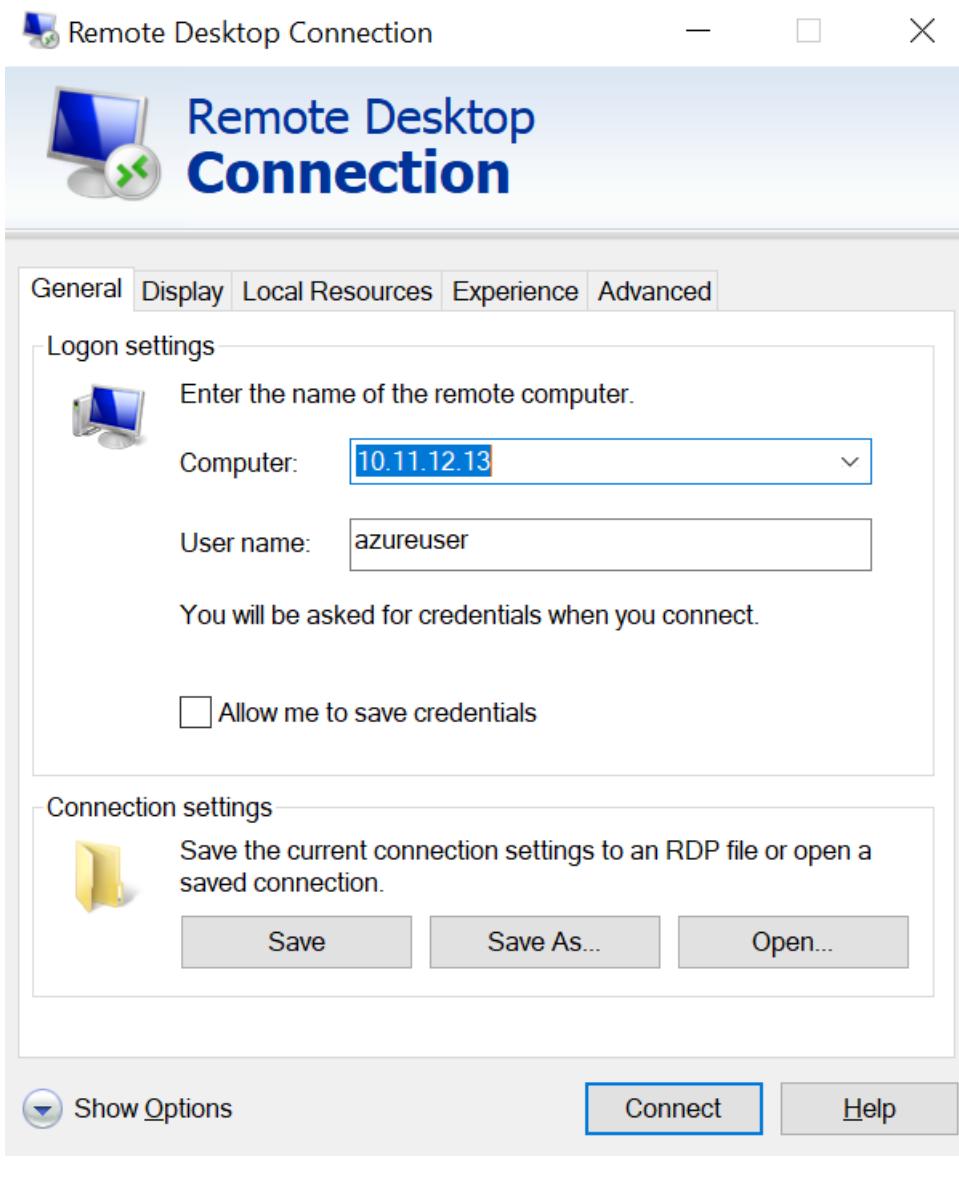
- [Azure CLI](#)
- [Azure PowerShell](#)

The following example creates a network security group rule with `az vm open-port` on port 3389. From the Azure CLI, not the SSH session to your VM, open the following network security group rule:

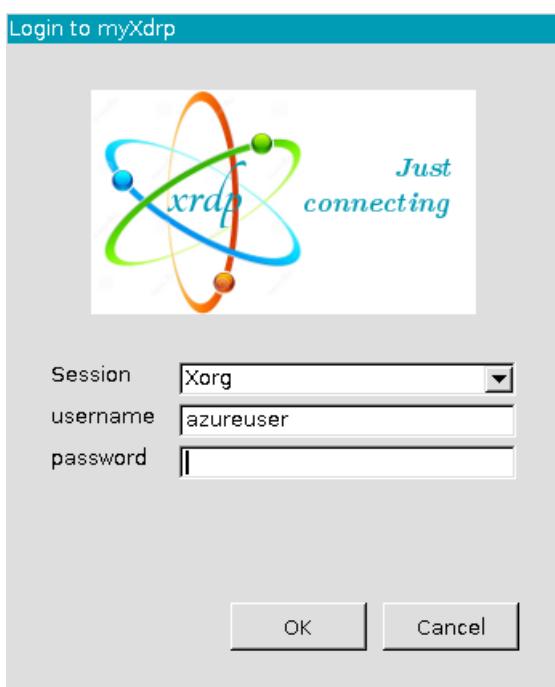
```
az vm open-port --resource-group myResourceGroup --name myVM --port 3389
```

## Connect your Linux VM with a Remote Desktop client

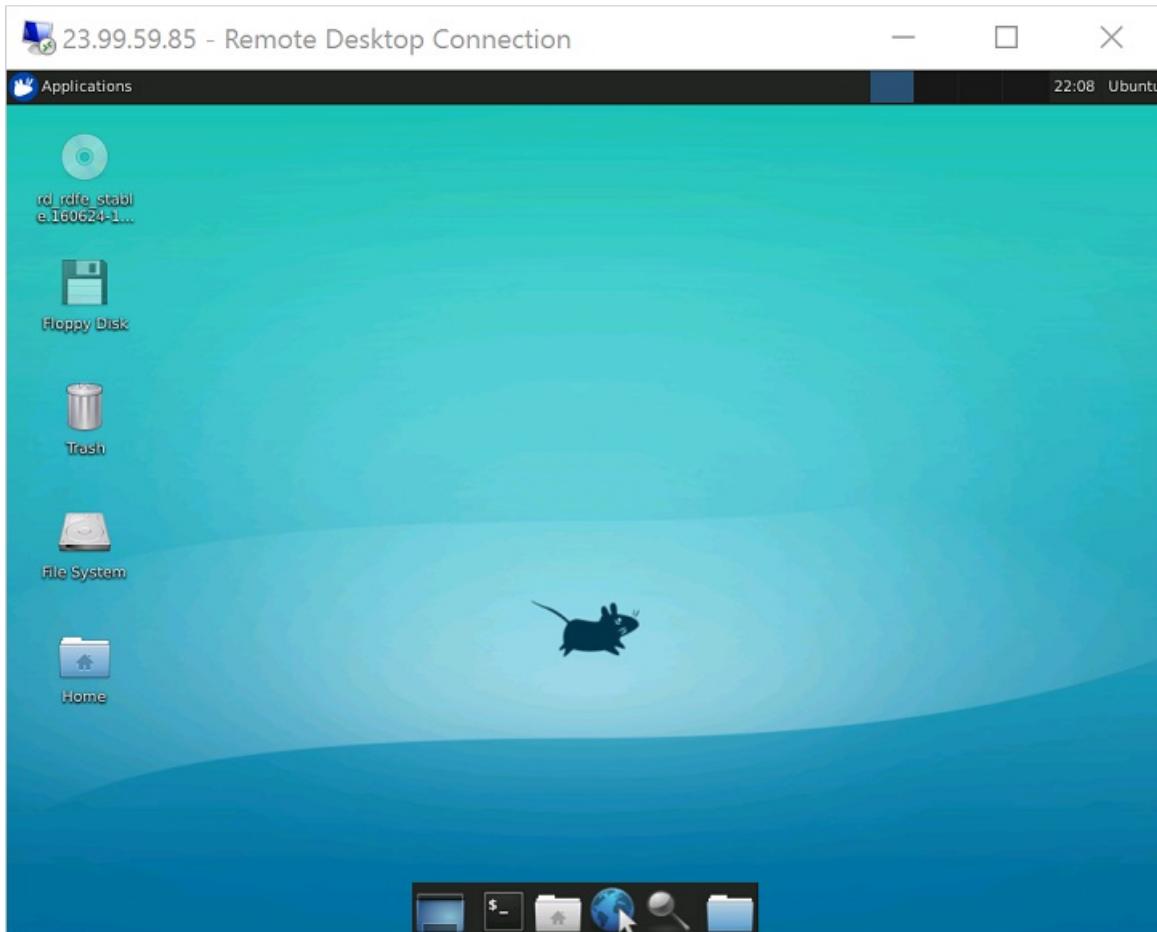
Open your local remote desktop client and connect to the IP address or DNS name of your Linux VM.



Enter the username and password for the user account on your VM as follows:



After authenticating, the xfce desktop environment will load and look similar to the following example:



If your local RDP client uses network level authentication (NLA), you may need to disable that connection setting. XRDП does not currently support NLA. You can also look at alternative RDP solutions that do support NLA, such as [FreeRDP](#).

## Troubleshoot

If you cannot connect to your Linux VM using a Remote Desktop client, use `netstat` on your Linux VM to verify that your VM is listening for RDP connections as follows:

```
sudo netstat -plnt | grep rdp
```

The following example shows the VM listening on TCP port 3389 as expected:

```
tcp      0      0      127.0.0.1:3350      0.0.0.0:*      LISTEN      53192/xrdp-sesman
tcp      0      0      0.0.0.0:3389      0.0.0.0:*      LISTEN      53188/xrdp
```

If the `xrdp-sesman` service is not listening, on an Ubuntu VM restart the service as follows:

```
sudo service xrdp restart
```

Review logs in `/var/log` on your Ubuntu VM for indications as to why the service may not be responding. You can also monitor the syslog during a remote desktop connection attempt to view any errors:

```
tail -f /var/log/syslog
```

Other Linux distributions such as Red Hat Enterprise Linux and SUSE may have different ways to restart services and alternate log file locations to review.

If you do not receive any response in your remote desktop client and do not see any events in the system log, this behavior indicates that remote desktop traffic cannot reach the VM. Review your network security group rules to ensure that you have a rule to permit TCP on port 3389. For more information, see [Troubleshoot application connectivity issues](#).

## Next steps

For more information about creating and using SSH keys with Linux VMs, see [Create SSH keys for Linux VMs in Azure](#).

For information on using SSH from Windows, see [How to use SSH keys with Windows](#).

# How to connect using Remote Desktop and sign on to an Azure virtual machine running Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

You can create a remote desktop connection to a virtual machine (VM) running Windows in Azure.

To connect to a Windows VM from a Mac, you will need to install an RDP client for Mac such as [Microsoft Remote Desktop](#).

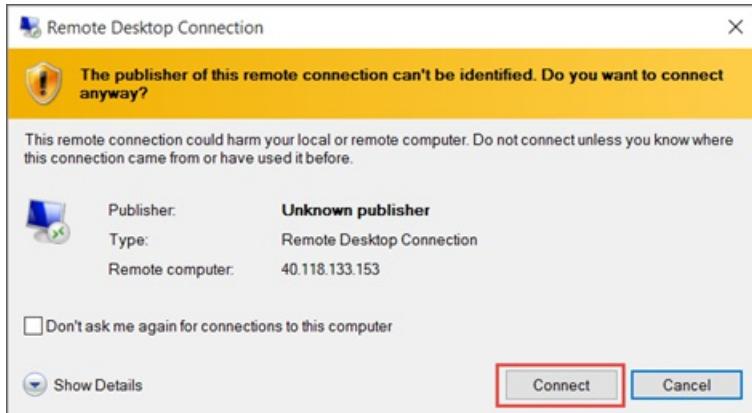
## Prerequisites

- In order to connect to a Windows Virtual Machine via RDP you need TCP connectivity to the machine on the port where Remote Desktop service is listening (3389 by default). You can validate an appropriate port is open for RDP using the troubleshooter or by checking manually in your VM settings. To check if the TCP port is open (assuming default):
  1. On the page for the VM, select **Networking** from the left menu.
  2. On the **Networking** page, check to see if there is a rule which allows TCP on port 3389 from the IP address of the computer you are using to connect to the VM. If the rule exists, you can move to the next section.
  3. If there isn't a rule, add one by selecting **Add Inbound port rule**.
  4. From the **Service** dropdown select **RDP**.
  5. Edit **Priority** and **Source** if necessary
  6. For **Name**, type *Port\_3389*
  7. When finished, select **Add**
  8. You should now have an RDP rule in the table of inbound port rules.
- Your VM must have a public IP address. To check if your VM has a public IP address, select **Overview** from the left menu and look at the **Networking** section. If you see an IP address next to **Public IP address**, then your VM has a public IP. To learn more about adding a public IP address to an existing VM, see [Associate a public IP address to a virtual machine](#)
- Verify your VM is running. On the Overview tab, in the essentials section, verify the status of the VM is **Running**. To start the VM, select **Start** at the top of the page.

## Connect to the virtual machine

1. Go to the [Azure portal](#) to connect to a VM. Search for and select **Virtual machines**.
2. Select the virtual machine from the list.
3. At the beginning of the virtual machine page, select **Connect**.
4. On the **Connect to virtual machine** page, select **RDP**, and then select the appropriate **IP address** and **Port number**. In most cases, the default IP address and port should be used. Select **Download RDP File**. If the VM has a just-in-time policy set, you first need to select the **Request access** button to request access before you can download the RDP file. For more information about the just-in-time policy, see [Manage virtual machine access using the just in time policy](#).
5. Open the downloaded RDP file and select **Connect** when prompted. You will get a warning that the

.rdp file is from an unknown publisher. This is expected. In the Remote Desktop Connection window, select **Connect** to continue.



6. In the Windows Security window, select **More choices** and then **Use a different account**. Enter the credentials for an account on the virtual machine and then select **OK**.

**Local account:** This is usually the local account user name and password that you specified when you created the virtual machine. In this case, the domain is the name of the virtual machine and it is entered as *vmname\username*.

**Domain joined VM:** If the VM belongs to a domain, enter the user name in the format *Domain\Username*. The account also needs to either be in the Administrators group or have been granted remote access privileges to the VM.

**Domain controller:** If the VM is a domain controller, enter the user name and password of a domain administrator account for that domain.

7. Select **Yes** to verify the identity of the virtual machine and finish logging on.



#### TIP

If the **Connect** button in the portal is grayed-out and you are not connected to Azure via an [Express Route](#) or [Site-to-Site VPN](#) connection, you will need to create and assign your VM a public IP address before you can use RDP. For more information, see [Public IP addresses in Azure](#).

## Connect to the virtual machine using PowerShell

If you are using PowerShell and have the Azure PowerShell module installed you may also connect using the `Get-AzRemoteDesktopFile` cmdlet, as shown below.

This example will immediately launch the RDP connection, taking you through similar prompts as above.

```
Get-AzRemoteDesktopFile -ResourceGroupName "RgName" -Name "VmName" -Launch
```

You may also save the RDP file for future use.

```
Get-AzRemoteDesktopFile -ResourceGroupName "RgName" -Name "VmName" -LocalPath "C:\Path\to\folder"
```

## Next steps

If you have difficulty connecting, see [Troubleshoot Remote Desktop connections](#).

# How to connect using Secure Shell (SSH) and sign on to an Azure virtual machine running Windows

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

The [Win32 OpenSSH](#) project makes remote connectivity using Secure Shell ubiquitous by providing native support in Windows. The capability is provided in Windows Server version 2019 and later, and can be added to older versions of Windows using a virtual machine (VM) extension.

The examples below use variables. You can set variables in your environment as follows.

SHELL	EXAMPLE
Bash/ZSH	myResourceGroup='resGroup10'
PowerShell	\$myResourceGroup='resGroup10'

## Enable SSH

First, you will need to enable SSH in your Windows machine.

### Windows Server 2019 and newer

Following the Windows Server documentation page [Get started with OpenSSH](#), run the command

```
Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
```

to enable the built-in capability, start the service, and open the Windows Firewall port.

You can use the Azure RunCommand extension to complete this task.

- [Azure CLI](#)
- [Azure PowerShell](#)
- [ARM template](#)
- [Bicep](#)

```
az vm run-command invoke -g $myResourceGroup -n $myVM --command-id RunPowerShellScript --scripts "Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0"
```

### Windows Server 2016 and older

- Deploy the SSH extension for Windows. The extension provides an automated installation of the Win32 OpenSSH solution, similar to enabling the capability in newer versions of Windows. Use the following examples to deploy the extension.
- [Azure CLI](#)
  - [Azure PowerShell](#)
  - [ARM template](#)
  - [Bicep](#)

```
az vm extension set --resource-group $myResourceGroup --vm-name $myVM --name WindowsOpenSSH --publisher Microsoft.Azure.OpenSSH --version 3.0
```

## Open TCP port

Ensure the appropriate port (by default, TCP 22) is open to allow connectivity to the VM.

- [Azure CLI](#)
- [Azure PowerShell](#)
- [ARM template](#)
- [Bicep](#)

```
az network nsg rule create -g $myResourceGroup --nsg-name $myNSG -n allow-SSH --priority 1000 --source-address-prefixes 208.130.28.4/32 --destination-port-ranges 22 --protocol TCP
```

- Your VM must have a public IP address. To check if your VM has a public IP address, select **Overview** from the left menu and look at the **Networking** section. If you see an IP address next to **Public IP address**, then your VM has a public IP. To learn more about adding a public IP address to an existing VM, see [Associate a public IP address to a virtual machine](#)
- Verify your VM is running. On the Overview tab, in the essentials section, verify the status of the VM is Running. To start the VM, select **Start** at the top of the page.

## Authentication

You can authenticate to Windows machines using either username and password or SSH keys. Azure does not support provisioning public keys to Windows machines automatically, however you can copy the key using the RunCommand extension.

## Overview of SSH and keys

SSH is an encrypted connection protocol that provides secure sign-ins over unsecured connections. Although SSH provides an encrypted connection, using passwords with SSH connections still leaves the VM vulnerable to brute-force attacks. We recommend connecting to a VM over SSH using a public-private key pair, also known as *SSH keys*.

- The *public key* is placed on your VM.
- The *private key* remains on your local system. Protect this private key. Do not share it.

When you use an SSH client to connect to your VM (which has the public key), the remote VM tests the client to make sure it has the correct private key. If the client has the private key, it's granted access to the VM.

Depending on your organization's security policies, you can reuse a single public-private key pair to access multiple Azure VMs and services. You do not need a separate pair of keys for each VM or service you wish to access.

Your public key can be shared with anyone, but only you (or your local security infrastructure) should have access to your private key.

## Supported SSH key formats

Azure currently supports SSH protocol 2 (SSH-2) RSA public-private key pairs with a minimum length of 2048

bits. Other key formats such as ED25519 and ECDSA are not supported.

### Copy a public key using the RunCommand extension.

The RunCommand extension provides an easy solution to copying a public key into Windows machines and making sure the file has correct permissions.

- [Azure CLI](#)
- [Azure PowerShell](#)
- [ARM template](#)
- [Bicep](#)

```
az vm run-command invoke -g $myResourceGroup -n $myVM --command-id RunPowerShellScript --scripts  
"MYPUBLICKEY | Add-Content 'C:\ProgramData\ssh\administratorsAuthorized_keys';icacls.exe  
'C:\ProgramData\ssh\administratorsAuthorized_keys' /inheritance:r /grant 'Administrators:F' /grant  
'SYSTEM:F'"
```

## Connect using Az CLI

Connect to Windows machines using [Az SSH](#) commands.

```
az ssh vm -g $myResourceGroup -n $myVM --local-user $myUsername
```

It is also possible to create a network tunnel for specific TCP ports through the SSH connection. A good use case for this is Remote Desktop which defaults to port 3389.

```
az ssh vm -g $myResourceGroup -n $myVM --local-user $myUsername -- -L 3389:localhost:3389
```

### Connect from Azure portal

1. Go to the [Azure portal](#) to connect to a VM. Search for and select **Virtual machines**.
2. Select the virtual machine from the list.
3. Select **Connect** from the left menu.
4. Select the **SSH** tab. If the VM has a just-in-time policy set, you first need to select the **Request access** button to request access before you can download the RDP file. For more information about the just-in-time policy, see [Manage virtual machine access using the just in time policy](#).

## Next steps

Learn how to transfer files to an existing VM, see [Use SCP to move files to and from a VM](#).

# Setting up WinRM access for Virtual Machines in Azure Resource Manager

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Here are the steps you need to take to set up a VM with WinRM connectivity

1. Create a Key Vault
2. Create a self-signed certificate
3. Upload your self-signed certificate to Key Vault
4. Get the URL for your self-signed certificate in the Key Vault
5. Reference your self-signed certificates URL while creating a VM

## Step 1: Create a Key Vault

You can use the below command to create the Key Vault

```
New-AzKeyVault -VaultName "<vault-name>" -ResourceGroupName "<rg-name>" -Location "<vault-location>" -EnabledForDeployment -EnabledForTemplateDeployment
```

## Step 2: Create a self-signed certificate

You can create a self-signed certificate using this PowerShell script

```
$certificateName = "somename"

$thumbprint = (New-SelfSignedCertificate -DnsName $certificateName -CertStoreLocation Cert:\CurrentUser\My -KeySpec KeyExchange).Thumbprint

$cert = (Get-ChildItem -Path cert:\CurrentUser\My\$thumbprint)

$password = Read-Host -Prompt "Please enter the certificate password." -AsSecureString

Export-PfxCertificate -Cert $cert -FilePath ".\$certificateName.pfx" -Password $password
```

## Step 3: Upload your self-signed certificate to the Key Vault

Before uploading the certificate to the Key Vault created in step 1, it needs to be converted into a format the Microsoft.Compute resource provider will understand. The below PowerShell script will allow you to do that

```

$fileName = "<Path to the .pfx file>"
$fileContentBytes = Get-Content $fileName -Encoding Byte
$fileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)
[System.Collections.HashTable]$TableForJSON = @{
    "data"      = $fileContentEncoded;
    "dataType" = "pfx";
    "password" = "<password>";
}
[System.String]$jsonObject = $TableForJSON | ConvertTo-Json
$encoding = [System.Text.Encoding]::UTF8
$jsonEncoded = [System.Convert]::ToBase64String($encoding.GetBytes($jsonObject))
$secret = ConvertTo-SecureString -String $jsonEncoded -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName "<vault name>" -Name "<secret name>" -SecretValue $secret

```

## Step 4: Get the URL for your self-signed certificate in the Key Vault

The Microsoft.Compute resource provider needs a URL to the secret inside the Key Vault while provisioning the VM. This enables the Microsoft.Compute resource provider to download the secret and create the equivalent certificate on the VM.

### **NOTE**

The URL of the secret needs to include the version as well. An example URL looks like below

<https://contosovault.vault.azure.net:443/secrets/contososecret/01h9db0df2cd4300a20ence585a6s7ve>

### **Templates**

You can get the link to the URL in the template using the below code

```

"certificateUrl": "[reference(resourceId(resourceGroup().name, 'Microsoft.KeyVault/vaults/secrets', '<vault-name>', '<secret-name>'), '2015-06-01').secretUriWithVersion]"

```

### **PowerShell**

You can get this URL using the below PowerShell command

```
$secretURL = (Get-AzKeyVaultSecret -VaultName "<vault name>" -Name "<secret name>").Id
```

## Step 5: Reference your self-signed certificates URL while creating a VM

### **Azure Resource Manager Templates**

While creating a VM through templates, the certificate gets referenced in the secrets section and the winRM section as below:

```

"osProfile": {
    ...
    "secrets": [
        {
            "sourceVault": {
                "id": "<resource id of the Key Vault containing the secret>"
            },
            "vaultCertificates": [
                {
                    "certificateUrl": "<URL for the certificate you got in Step 4>",
                    "certificateStore": "<Name of the certificate store on the VM>"
                }
            ]
        }
    ],
    "windowsConfiguration": {
        ...
        "winRM": {
            "listeners": [
                {
                    "protocol": "http"
                },
                {
                    "protocol": "https",
                    "certificateUrl": "<URL for the certificate you got in Step 4>"
                }
            ]
        },
        ...
    }
},

```

A sample template for the above can be found here at [vm-winrm-keyvault-windows](#)

Source code for this template can be found on [GitHub](#)

#### PowerShell

```

$vm = New-AzVMConfig -VMName "<VM name>" -VMSize "<VM Size>"
$credential = Get-Credential
$secretURL = (Get-AzKeyVaultSecret -VaultName "<vault name>" -Name "<secret name>").Id
$vm = Set-AzVMOperatingSystem -VM $vm -Windows -ComputerName "<Computer Name>" -Credential $credential -
WinRMHttp -WinRMHttps -ProvisionVMAgent -WinRMCertificateUrl $secretURL
$sourceVaultId = (Get-AzKeyVault -ResourceGroupName "<Resource Group name>" -VaultName "<Vault
Name>").ResourceId
$CertificateStore = "My"
$vm = Add-AzVMSecret -VM $vm -SourceVaultId $sourceVaultId -CertificateStore $CertificateStore -
CertificateUrl $secretURL

```

## Step 6: Connecting to the VM

Before you can connect to the VM, you'll need to make sure your machine is configured for WinRM remote management. Start PowerShell as an administrator and execute the below command to make sure you're set up.

```
Enable-PSRemoting -Force
```

#### NOTE

You might need to make sure the WinRM service is running if the above does not work. You can do that using

```
Get-Service WinRM
```

Once the setup is done, you can connect to the VM using the below command

```
Enter-PSSession -ConnectionUri https://<public-ip-dns-of-the-vm>:5986 -Credential $cred -SessionOption (New-PSSessionOption -SkipCACheck -SkipCNCheck -SkipRevocationCheck) -Authentication Negotiate
```

# Time sync for Linux VMs in Azure

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

Time sync is important for security and event correlation. Sometimes it is used for distributed transactions implementation. Time accuracy between multiple computer systems is achieved through synchronization. Synchronization can be affected by multiple things, including reboots and network traffic between the time source and the computer fetching the time.

Azure is backed by infrastructure running Windows Server 2016. Windows Server 2016 has improved algorithms used to correct time and condition the local clock to synchronize with UTC. The Windows Server 2016 Accurate Time feature greatly improved how the VMCTimeSync service that governs VMs with the host for accurate time. Improvements include more accurate initial time on VM start or VM restore and interrupt latency correction.

## NOTE

For a quick overview of Windows Time service, take a look at this [high-level overview video](#).

For more information, see [Accurate time for Windows Server 2016](#).

## Overview

Accuracy for a computer clock is gauged on how close the computer clock is to the Coordinated Universal Time (UTC) time standard. UTC is defined by a multinational sample of precise atomic clocks that can only be off by one second in 300 years. But, reading UTC directly requires specialized hardware. Instead, time servers are synced to UTC and are accessed from other computers to provide scalability and robustness. Every computer has time synchronization service running that knows what time servers to use and periodically checks if computer clock needs to be corrected and adjusts time if needed.

Azure hosts are synchronized to internal Microsoft time servers that take their time from Microsoft-owned Stratum 1 devices, with GPS antennas. Virtual machines in Azure can either depend on their host to pass the accurate time (*host time*) on to the VM or the VM can directly get time from a time server, or a combination of both.

On stand-alone hardware, the Linux OS only reads the host hardware clock on boot. After that, the clock is maintained using the interrupt timer in the Linux kernel. In this configuration, the clock will drift over time. In newer Linux distributions on Azure, VMs can use the VMCTimeSync provider, included in the Linux integration services (LIS), to query for clock updates from the host more frequently.

Virtual machine interactions with the host can also affect the clock. During [memory preserving maintenance](#), VMs are paused for up to 30 seconds. For example, before maintenance begins the VM clock shows 10:00:00 AM and lasts 28 seconds. After the VM resumes, the clock on the VM would still show 10:00:00 AM, which would be 28 seconds off. To correct for this, the VMCTimeSync service monitors what is happening on the host and updates the time-of-day clock in Linux VMs to compensate.

Without time synchronization working, the clock on the VM would accumulate errors. When there is only one VM, the effect might not be significant unless the workload requires highly accurate timekeeping. But in most cases, we have multiple, interconnected VMs that use time to track transactions and the time needs to be consistent throughout the entire deployment. When time between VMs is different, you could see the following

effects:

- Authentication will fail. Security protocols like Kerberos or certificate-dependent technology rely on time being consistent across the systems.
- It's very hard to figure out what has happened in a system if logs (or other data) don't agree on time. The same event would look like it occurred at different times, making correlation difficult.
- If clock is off, the billing could be calculated incorrectly.

## Configuration options

Time sync requires that a time sync service be running in the Linux VM, plus a source of accurate time information against which to synchronize. Typically ntpd or chronyd is used as the time sync service, though there are other open source time sync services that can be used as well. The source of accurate time information can be the Azure host or an external time service that is accessed over the public internet. By itself, the VMICTimeSync service does not provide ongoing time sync between the Azure host and a Linux VM except after pauses for host maintenance as described above.

Historically, most Azure Marketplace images with Linux have been configured in one of two ways:

- No time sync service is running by default
- ntpd is running as the time sync service, and synchronizing against an external NTP time source that is accessed over the network. For example, Ubuntu 18.04 LTS Marketplace images use [ntp.ubuntu.com](http://ntp.ubuntu.com).

To confirm ntpd is synchronizing correctly, run the `ntpq -p` command.

Some Azure Marketplace images with Linux are being changed to use chronyd as the time sync service, and chronyd is configured to synchronize against the Azure host rather than an external NTP time source. The Azure host time is usually the best time source to synchronize against, as it is maintained very accurately and reliably, and is accessible without the variable network delays inherent in accessing an external NTP time source over the public internet.

The VMICTimeSync is used in parallel and provides two functions:

- Immediately updates the Linux VM time-of-day clock after a host maintenance event
- Instantiates an IEEE 1588 Precision Time Protocol (PTP) hardware clock source as a /dev/ptp device that provides the accurate time-of-day from the Azure host. Chronyd can be configured to synchronize against this time source (which is the default configuration in the newest Linux images). Linux distributions with kernel version 4.11 or later (or version 3.10.0-693 or later for RHEL/CentOS 7) support the /dev/ptp device. For earlier kernel versions that do not support /dev/ptp for Azure host time, only synchronization against an external time source is possible.

Of course, the default configuration can be changed. An older image that is configured to use ntpd and an external time source can be changed to use chronyd and the /dev/ptp device for Azure host time. Similarly, an image using Azure host time via a /dev/ptp device can be configured to use an external NTP time source if required by your application or workload.

## Tools and resources

There are some basic commands for checking your time synchronization configuration. Documentation for Linux distribution will have more details on the best way to configure time synchronization for that distribution.

### Integration services

Check to see if the integration service (hv\_utils) is loaded.

```
lsmod | grep hv_utils
```

You should see something similar to this:

```
hv_utils           24418  0
hv_vmbus          397185  7
hv_balloon,hyperv_keyboard,hv_netvsc,hid_hyperv,hv_utils,hyperv_fb,hv_storvsc
```

## Check for PTP Clock Source

With newer versions of Linux, a Precision Time Protocol (PTP) clock source corresponding to the Azure host is available as part of the VMICTimeSync provider. On older versions of Red Hat Enterprise Linux or CentOS 7.x the [Linux Integration Services](#) can be downloaded and used to install the updated driver. When the PTP clock source is available, the Linux device will be of the form /dev/ptpx.

See which PTP clock sources are available.

```
ls /sys/class/ptp
```

In this example, the value returned is *ptp0*, so we use that to check the clock name. To verify the device, check the clock name.

```
cat /sys/class/ptp/ptp0/clock_name
```

This should return `hyperv`, meaning the Azure host.

In Linux VMs with Accelerated Networking enabled, you may see multiple PTP devices listed because the Mellanox mlx5 driver also creates a /dev/ptp device. Because the initialization order can be different each time Linux boots, the PTP device corresponding to the Azure host might be `/dev/ptp0` or it might be `/dev/ptp1`, which makes it difficult to configure `chronyd` with the correct clock source. To solve this problem, the most recent Linux images have a `udev` rule that creates the symlink `/dev/ptp_hyperv` to whichever `/dev/ptp` entry corresponds to the Azure host. Chrony should be configured to use this symlink instead of `/dev/ptp0` or `/dev/ptp1`.

## chrony

On Ubuntu 19.10 and later versions, Red Hat Enterprise Linux, and CentOS 8.x, `chrony` is configured to use a PTP source clock. Instead of chrony, older Linux releases use the Network Time Protocol daemon (`ntpd`), which doesn't support PTP sources. To enable PTP in those releases, chrony must be manually installed and configured (in `chrony.conf`) by using the following statement:

```
refclock PHC /dev/ptp_hyperv poll 3 dpoll -2 offset 0 stratum 2
```

If the `/dev/ptp_hyperv` symlink is available, use it instead of `/dev/ptp0` to avoid any confusion with the `/dev/ptp` device created by the Mellanox mlx5 driver.

Stratum information isn't automatically conveyed from the Azure host to the Linux guest. The preceding configuration line specifies that the Azure host time source is to be treated as Stratum 2, which in turn causes the Linux guest to report itself as Stratum 3. You can change the stratum setting in the configuration line if you want the Linux guest to report itself differently.

By default, `chronyd` accelerates or slows the system clock to fix any time drift. If the drift becomes too big, `chrony` fails to fix the drift. To overcome this, the `makestep` parameter in `/etc/chrony.conf` can be changed to

force a time sync if the drift exceeds the threshold specified.

```
makestep 1.0 -1
```

Here, chrony will force a time update if the drift is greater than 1 second. To apply the changes restart the chronyd service:

```
sudo systemctl restart chronyd
```

For more information about Ubuntu and NTP, see [Time Synchronization](#).

For more information about Red Hat and NTP, see [Configure NTP](#).

For more information about chrony, see [Using chrony](#).

## systemd

On SUSE and Ubuntu releases before 19.10, time sync is configured using [systemd](#). For more information about Ubuntu, see [Time Synchronization](#). For more information about SUSE, see Section 4.5.8 in [SUSE Linux Enterprise Server 12 SP3 Release Notes](#).

## Next steps

For more information, see [Accurate time for Windows Server 2016](#).

# Time sync for Windows VMs in Azure

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Time sync is important for security and event correlation. Sometimes it is used for distributed transactions implementation. Time accuracy between multiple computer systems is achieved through synchronization. Synchronization can be affected by multiple things, including reboots and network traffic between the time source and the computer fetching the time.

Azure is now backed by infrastructure running Windows Server 2016. Windows Server 2016 has improved algorithms used to correct time and condition the local clock to synchronize with UTC. Windows Server 2016 also improved the VMCTimeSync service that governs how VMs sync with the host for accurate time. Improvements include more accurate initial time on VM start or VM restore and interrupt latency correction for samples provided to Windows Time (W32time).

## NOTE

For a quick overview of Windows Time service, take a look at this [high-level overview video](#).

For more information, see [Accurate time for Windows Server 2016](#).

## Overview

Accuracy for a computer clock is gauged on how close the computer clock is to the Coordinated Universal Time (UTC) time standard. UTC is defined by a multinational sample of precise atomic clocks that can only be off by one second in 300 years. But, reading UTC directly requires specialized hardware. Instead, time servers are synced to UTC and are accessed from other computers to provide scalability and robustness. Every computer has time synchronization service running that knows what time servers to use and periodically checks if computer clock needs to be corrected and adjusts time if needed.

Azure hosts are synchronized to internal Microsoft time servers that take their time from Microsoft-owned Stratum 1 devices, with GPS antennas. Virtual machines in Azure can either depend on their host to pass the accurate time (*host time*) on to the VM or the VM can directly get time from a time server, or a combination of both.

Virtual machine interactions with the host can also affect the clock. During [memory preserving maintenance](#), VMs are paused for up to 30 seconds. For example, before maintenance begins the VM clock shows 10:00:00 AM and lasts 28 seconds. After the VM resumes, the clock on the VM would still show 10:00:00 AM, which would be 28 seconds off. To correct for this, the VMCTimeSync service monitors what is happening on the host and prompts for changes to happen on the VMs to compensate.

The VMCTimeSync service operates in either sample or sync mode and will only influence the clock forward. In sample mode, which requires W32time to be running, the VMCTimeSync service polls the host every 5 seconds and provides time samples to W32time. Approximately every 30 seconds, the W32time service takes the latest time sample and uses it to influence the guest's clock. Sync mode activates if a guest has been resumed or if a guest's clock drifts more than 5 seconds behind the host's clock. In cases where the W32time service is properly running, the latter case should never happen.

Without time synchronization working, the clock on the VM would accumulate errors. When there is only one VM, the effect might not be significant unless the workload requires highly accurate timekeeping. But in most

cases, we have multiple, interconnected VMs that use time to track transactions and the time needs to be consistent throughout the entire deployment. When time between VMs is different, you could see the following effects:

- Authentication will fail. Security protocols like Kerberos or certificate-dependent technology rely on time being consistent across the systems.
- It's very hard to figure out what has happened in a system if logs (or other data) don't agree on time. The same event would look like it occurred at different times, making correlation difficult.
- If clock is off, the billing could be calculated incorrectly.

The best results for Windows deployments are achieved by using Windows Server 2016 as the guest operating system, which ensures you can use the latest improvements in time synchronization.

## Configuration options

There are three options for configuring time sync for your Windows VMs hosted in Azure:

- Host time and time.windows.com. This is the default configuration used in Azure Marketplace images.
- Host-only.
- Use another, external time server with or without using host time. For this option follow the [Configure Azure Windows VMs with External NTP Source](#) guide.

### Use the default

By default Windows OS VM images are configured for w32time to sync from two sources:

- The NtpClient provider, which gets information from time.windows.com.
- The VMICTimeSync service, used to communicate the host time to the VMs and make corrections after the VM is paused for maintenance. Azure hosts use Microsoft-owned Stratum 1 devices to keep accurate time.

w32time would prefer the time provider in the following order of priority: stratum level, root delay, root dispersion, time offset. In most cases, w32time on an Azure VM would prefer host time due to evaluation it would do to compare both time sources.

For domain joined machines the domain itself establishes time sync hierarchy, but the forest root still needs to take time from somewhere and the following considerations would still hold true.

### Host-only

Because time.windows.com is a public NTP server, syncing time with it requires sending traffic over the internet, varying packet delays can negatively affect quality of the time sync. Removing time.windows.com by switching to host-only sync can sometimes improve your time sync results.

Switching to host-only time sync makes sense if you experience time sync issues using the default configuration. Try out the host-only sync to see if that would improve the time sync on VM.

### External time server

If you have specific time sync requirements, there is also an option of using external time servers. External time servers can provide specific time, which can be useful for test scenarios, ensuring time uniformity with machines hosted in non-Microsoft datacenters, or handling leap seconds in a special way.

You can combine external servers with the VMICTimeSync service and VMICTimeProvider to provide results similar to the default configuration.

## Check your configuration

Check if the NtpClient time provider is configured to use explicit NTP servers (NTP) or domain time sync (NT5DS).

```
w32tm /dumpreg /subkey:Parameters | findstr /i "type"
```

If the VM is using NTP, you will see the following output:

Value Name	Value Type	Value Data
Type	REG_SZ	NTP

To see what time server the NtpClient time provider is using, at an elevated command prompt type:

```
w32tm /dumpreg /subkey:Parameters | findstr /i "ntpserver"
```

If the VM is using the default, the output will look like this:

NtpServer	REG_SZ	time.windows.com,0x8
-----------	--------	----------------------

To see what time provider is being used currently.

```
w32tm /query /source
```

Here is the output you could see and what it would mean:

- **time.windows.com** - in the default configuration, w32time would get time from time.windows.com. The time sync quality depends on internet connectivity to it and is affected by packet delays. This is the usual output you would get on a physical machine.
- **VM IC Time Synchronization Provider** - the VM is syncing time from the host. This is the usual output you would get on a virtual machine running on Azure.
- *Your domain server* - the current machine is in a domain and the domain defines the time sync hierarchy.
- *Some other server* - w32time was explicitly configured to get the time from that another server. Time sync quality depends on this time server quality.
- **Local CMOS Clock** - clock is unsynchronized. You can get this output if w32time hasn't had enough time to start after a reboot or when all the configured time sources are not available.

## Opt in for host-only time sync

Azure is constantly working on improving time sync on hosts and can guarantee that all the time sync infrastructure is collocated in Microsoft-owned datacenters. If you have time sync issues with the default setup that prefers to use time.windows.com as the primary time source, you can use the following commands to opt in to host-only time sync.

Mark the VMIC provider as enabled.

```
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\TimeProviders\VMICTimeProvider /v Enabled /t REG_DWORD /d 1 /f
```

Mark the NTPClient provider as disabled.

```
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\TimeProviders\NtpClient /v Enabled /t REG_DWORD /d 0 /f
```

Restart the w32time Service.

```
net stop w32time && net start w32time
```

## Windows Server 2012 and R2 VMs

Windows Server 2012 and Windows Server 2012 R2 have different default settings for time sync. The w32time by default is configured in a way that prefers low overhead of the service over precise time.

If you want to move your Windows Server 2012 and 2012 R2 deployments to use the newer defaults that prefer precise time, you can apply the following settings.

Update the w32time poll and update intervals to match Windows Server 2016 settings.

```
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\Config /v MinPollInterval /t REG_DWORD /d 6 /f  
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\Config /v MaxPollInterval /t REG_DWORD /d 10 /f  
reg add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\Config /v UpdateInterval /t REG_DWORD /d 100 /f  
w32tm /config /update
```

For `w32time` to be able to use the new poll intervals the NtpServers need to be marked as using them. If servers are annotated with the `0x1` bitflag mask, that would override this mechanism and `w32time` would use `SpecialPollInterval` instead. Make sure that specified NTP servers are either using the `0x8` flag or no flag at all:

Check what flags are being used for the NTP servers.

```
w32tm /dumpreg /subkey:Parameters | findstr /i "ntpserver"
```

## Next steps

Below are links to more details about the time sync:

- [Windows Time Service Tools and Settings](#)
- [Windows Server 2016 Improvements](#)
- [Accurate Time for Windows Server 2016](#)
- [Support boundary to configure the Windows Time service for high-accuracy environments](#)

# Configure Active Directory Windows Virtual Machines in Azure with External NTP Source

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows Virtual Machines

Use this guide to learn how to setup time synchronization for your Azure Windows Virtual Machines that belong to an Active Directory Domain with an external NTP source.

## Time Sync for Active Directory Windows Virtual Machines in Azure with External NTP Source

Time synchronization in Active Directory should be managed by only allowing the PDC to access an external time source or NTP Server. All other Domain Controllers would then sync time against the PDC. If your PDC is an Azure Virtual Machine follow these steps:

### NOTE

Due to Azure Security configurations, the following settings must be applied on the PDC using the **Local Group Policy Editor**.

To check current time source in your **PDC**, from an elevated command prompt run `w32tm /query /source` and note the output for later comparison.

1. From *Start* run `gpedit.msc`
2. Navigate to the *Global Configuration Settings* policy under *Computer Configuration -> Administrative Templates -> System -> Windows Time Service*.
3. Set it to *Enabled* and configure the *AnnounceFlags* parameter to 5.
4. Navigate to *Computer Settings -> Administrative Templates -> System -> Windows Time Service -> Time Providers*.
5. Double click the *Configure Windows NTP Client* policy and set it to *Enabled*, configure the parameter *NTPServer* to point to an IP address of a time server followed by `,0x9` for example: `131.107.13.100,0x9` and configure *Type* to NTP. For all the other parameters you can use the default values, or use custom ones according to your corporate needs.

### IMPORTANT

You must mark the VMIC provider as *Disabled* in the Local Registry. Remember that serious problems might occur if you modify the registry incorrectly. Therefore, make sure that you follow these steps carefully. For added protection, back up the registry before you modify it. Then, you can restore the registry if a problem occurs. For how to back up and restore the Windows Registry follow the steps below.

## Back up the registry manually

- Select Start, type `regedit.exe` in the search box, and then press Enter. If you are prompted for an administrator password or for confirmation, type the password or provide confirmation.
- In Registry Editor, locate and click the registry key or subkey that you want to back up.

- Select File -> Export.
- In the Export Registry File dialog box, select the location to which you want to save the backup copy, and then type a name for the backup file in the File name field.
- Select Save.

## Restore a manual backup

- Select Start, type regedit.exe, and then press Enter. If you are prompted for an administrator password or for confirmation, type the password or provide confirmation.
- In Registry Editor, click File -> Import.
- In the Import Registry File dialog box, select the location to which you saved the backup copy, select the backup file, and then click Open.

To mark the VMIC provider as *Disabled* from *Start* type *regedit.exe* -> In the *Registry Editor* navigate to *HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\w32time\TimeProviders* -> On key *VMICTimeProvider* set the value to 0

### NOTE

It can take up to 15 minutes for these changes to reflect in the system.

From an elevated command prompt rerun *w32tm /query /source* and compare the output to the one you noted at the beginning of the configuration. Now it will be set to the NTP Server you chose.

## GPO for Clients

Configure the following Group Policy Object to enable your clients to synchronize time with any Domain Controller in your Domain:

To check current time source in your client, from an elevated command prompt run *w32tm /query /source* and note the output for later comparison.

1. From a Domain Controller go to *Start* run *gpmc.msc*
2. Browse to the Forest and Domain where you want to create the GPO.
3. Create a new GPO, for example *Clients Time Sync*, in the container *Group Policy Objects*.
4. Right-click on the newly created GPO and Edit.
5. In the *Group Policy Management Editor* navigate to the *Configure Windows NTP Client* policy under *Computer Configuration -> Administrative Templates -> System -> Windows Time Service -> Time Providers*
6. Set it to *Enabled*, configure the parameter *NTPServer* to point to a Domain Controller in your Domain followed by *,0x8* for example: *DC1.contoso.com,0x8* and configure *Type* to NT5DS. For all the other parameters you can use the default values, or use custom ones according to your corporate needs.
7. Link the GPO to the Organizational Unit where your clients are located.

### IMPORTANT

In the the parameter *NTPServer* you can specify a list with all the Domain Controllers in your domain, like this:

*DC1.contoso.com,0x8 DC2.contoso.com,0x8 DC3.contoso.com,0x8*

From an elevated command prompt rerun *w32tm /query /source* and compare the output to the one you noted at the beginning of the configuration. Now it will be set to the Domain Controller that satisfied the client's authentication request.

## Next steps

Below are links to more details about the time sync:

- [Windows Time Service Tools and Settings](#)
- [Windows Server 2016 Improvements](#)
- [Accurate Time for Windows Server 2016](#)
- [Support boundary to configure the Windows Time service for high-accuracy environments](#)

# Run scripts in your VM by using Run Command

9/21/2022 • 2 minutes to read • [Edit Online](#)

Run Command uses the virtual machine (VM) agent to run scripts within an Azure Windows or Linux VM. You can use these scripts for general machine or application management. They can help you to quickly diagnose and remediate VM access and network issues and get the VM back to a good state. Scripts can be embedded in the properties or referenced to a pre published gallery script.

The original set of commands are action orientated. The updated set of commands, currently in Public Preview, are management orientated and enable you to run multiple scripts and has less restrictions. This article will explain the difference between the two sets of run commands and help you decide which set is the right one to use in your scenario.

## IMPORTANT

**Managed Run Command** is currently in public preview. This preview version is provided without a service-level agreement, and we don't recommend it for production workloads. Certain features might not be supported or might have constrained capabilities. For more information, see [Supplemental Terms of Use for Microsoft Azure Previews](#).

## When to use action or managed commands

The original set of commands are action orientated. You should consider using this set of commands for situations where you need to run:

- A small script to get a content from a VM
- A script to configure a VM (set registry keys, change configuration)
- A one time script for diagnostics

See [Action Run Commands for Linux](#) and [Action Run Commands for Windows](#) for available action commands and instructions on how to apply them.

The updated set of commands, currently in Public Preview, are management orientated. Consider using managed run commands if your needs align to the following examples:

- Script needs to run as part of VM deployment
- Recurrent script execution is needed
- Multiple scripts needs to execute sequentially
- Bootstrap a VM by running installation scripts
- Publish custom script to be shared and reused

See [Managed Run Command for Linux](#) and [Managed Run Command for Windows](#) to learn how to use them.

## Compare feature support

FEATURE SUPPORT	ACTION RUNCOMMAND	MANAGED RUNCOMMAND
ARM template	No, it's a POST action	Yes, it's a resource type
Long running	90 min limit	Customer specified timeout

FEATURE SUPPORT	ACTION RUNCOMMAND	MANAGED RUNCOMMAND
Execution account	System account / root	Customer specified user
Multiple run commands	Only one active	Multiple in parallel or sequenced
Large output	Limited to 4k (in status blob)	Uploaded to customer append blob
Progress tracking	Reports only final status	Reports progress and last 4k output during execution
Async execution	Goal state/provisioning waits for script to complete	Customer specified async flag if provisioning waits for the script
Virtual machine scale set support	Only on VM instance	Support virtual machine scale set model and scale out
SAS generation	No blob support	Automated, CRP generates SAS for customer blobs and manages them
Gallery (custom commands)	Only built-in commandIds	Customer can publish scripts and share them

## Next steps

Get started with [Managed Run Command for Linux](#) or [Managed Run Command for Windows](#).

# Run scripts in your Linux VM by using action Run Commands

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The Run Command feature uses the virtual machine (VM) agent to run shell scripts within an Azure Linux VM. You can use these scripts for general machine or application management. They can help you to quickly diagnose and remediate VM access and network issues and get the VM back to a good state.

## Benefits

You can access your virtual machines in multiple ways. Run Command can run scripts on your virtual machines remotely by using the VM agent. You use Run Command through the Azure portal, [REST API](#), or [Azure CLI](#) for Linux VMs.

This capability is useful in all scenarios where you want to run a script within a virtual machine. It's one of the only ways to troubleshoot and remediate a virtual machine that doesn't have the RDP or SSH port open because of network or administrative user configuration.

## Restrictions

The following restrictions apply when you're using Run Command:

- Output is limited to the last 4,096 bytes.
- The minimum time to run a script is about 20 seconds.
- Scripts run by default as an elevated user on Linux.
- You can run one script at a time.
- Scripts that prompt for information (interactive mode) are not supported.
- You can't cancel a running script.
- The maximum time a script can run is 90 minutes. After that, the script will time out.
- Outbound connectivity from the VM is required to return the results of the script.

### NOTE

To function correctly, Run Command requires connectivity (port 443) to Azure public IP addresses. If the extension doesn't have access to these endpoints, the scripts might run successfully but not return the results. If you're blocking traffic on the virtual machine, you can use [service tags](#) to allow traffic to Azure public IP addresses by using the `AzureCloud` tag.

## Available commands

This table shows the list of commands available for Linux VMs. You can use the `RunShellScript` command to run any custom script that you want. When you're using the Azure CLI or PowerShell to run a command, the value that you provide for the `--command-id` or `-CommandId` parameter must be one of the following listed values. When you specify a value that is not an available command, you receive this error:

The entity was not found in this Azure location

NAME	DESCRIPTION
RunShellScript	Runs a Linux shell script.
ifconfig	Gets the configuration of all network interfaces.

## Azure CLI

The following example uses the [az vm run-command](#) command to run a shell script on an Azure Linux VM.

```
az vm run-command invoke -g myResourceGroup -n myVm --command-id RunShellScript --scripts "apt-get update && apt-get install -y nginx"
```

### NOTE

To run commands as a different user, enter `sudo -u` to specify a user account.

## Azure portal

Go to a VM in the [Azure portal](#) and select **Run command** in the left menu, under **Operations**. You see a list of the available commands to run on the VM.

NAME	DESCRIPTION
RunShellScript	Executes a Linux shell script
ifconfig	List network configuration

Choose a command to run. Some of the commands might have optional or required input parameters. For those commands, the parameters are presented as text fields for you to provide the input values. For each command, you can view the script that's being run by expanding **View script**. **RunShellScript** is different from the other commands, because it allows you to provide your own custom script.

### NOTE

The built-in commands are not editable.

After you choose the command, select **Run** to run the script. After the script finishes, it returns the output and any errors in the output window. The following screenshot shows an example output from running the **ifconfig** command.

Run Command Script  
ifconfig

**Script execution complete**

**Details**  
Get the configuration of all network interfaces.

View script

**Parameters**

**ARGUMENTS** Default will be used

**Run**

**Output**

```
Enable succeeded:
[stdout]
eth0      Link encap:Ethernet  HWaddr 00:0d:3a:12:81:d1
          inet  addr:10.0.0.7   Bcast:10.0.0.255  Mask:255.255.255.0
          inet6 addr: fe80::20d:3aff:fe12:81d1/64 Scope:Link
                  UP BROADCAST RUNNING MULTICAST  MTU:1500 Metric:1
                  RX packets:2134524 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:1438287 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:2208675468 (2.2 GB)  TX bytes:400069292 (400.0 MB)

lo        Link encap:Local Loopback
          inet  addr:127.0.0.1   Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
                  UP LOOPBACK RUNNING  MTU:65536 Metric:1
                  RX packets:160 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:160 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:11840 (11.8 KB)  TX bytes:11840 (11.8 KB)

[stderr]
```

## PowerShell

The following example uses the [Invoke-AzVMRunCommand](#) cmdlet to run a PowerShell script on an Azure VM. The cmdlet expects the script referenced in the `-ScriptPath` parameter to be local to where the cmdlet is being run.

```
Invoke-AzVMRunCommand -ResourceGroupName '<myResourceGroup>' -Name '<myVMName>' -CommandId 'RunShellScript' -ScriptPath '<pathToScript>' -Parameter @{"arg1" = "var1";"arg2" = "var2"}
```

## Limiting access to Run Command

Listing the run commands or showing the details of a command requires the

`Microsoft.Compute/locations/runCommands/read` permission. The built-in [Reader](#) role and higher levels have this permission.

Running a command requires the `Microsoft.Compute/virtualMachines/runCommand/action` permission. The [Virtual Machine Contributor](#) role and higher levels have this permission.

You can use one of the [built-in roles](#) or create a [custom role](#) to use Run Command.

## Next steps

To learn about other ways to run scripts and commands remotely in your VM, see [Run scripts in your Linux VM](#).

# Run scripts in your Windows VM by using action Run Commands

9/21/2022 • 4 minutes to read • [Edit Online](#)

The Run Command feature uses the virtual machine (VM) agent to run PowerShell scripts within an Azure Windows VM. You can use these scripts for general machine or application management. They can help you to quickly diagnose and remediate VM access and network issues and get the VM back to a good state.

## Benefits

You can access your virtual machines in multiple ways. Run Command can run scripts on your virtual machines remotely by using the VM agent. You use Run Command through the Azure portal, [REST API](#), or [PowerShell](#) for Windows VMs.

This capability is useful in all scenarios where you want to run a script within a virtual machine. It's one of the only ways to troubleshoot and remediate a virtual machine that doesn't have the RDP or SSH port open because of improper network or administrative user configuration.

## Restrictions

The following restrictions apply when you're using Run Command:

- Output is limited to the last 4,096 bytes.
- The minimum time to run a script is about 20 seconds.
- Scripts run as System on Windows.
- One script at a time can run.
- Scripts that prompt for information (interactive mode) are not supported.
- You can't cancel a running script.
- The maximum time a script can run is 90 minutes. After that, it will time out.
- Outbound connectivity from the VM is required to return the results of the script.
- It is not recommended to run a script that will cause a stop or update of the VM Agent. This can let the extension in a Transitioning state, leading to a timeout.

### NOTE

To function correctly, Run Command requires connectivity (port 443) to Azure public IP addresses. If the extension doesn't have access to these endpoints, the scripts might run successfully but not return the results. If you're blocking traffic on the virtual machine, you can use [service tags](#) to allow traffic to Azure public IP addresses by using the `AzureCloud` tag.

The Run Command feature doesn't work if the VM agent status is NOT READY. Check the agent status in the VM's properties in the Azure portal.

## Available commands

This table shows the list of commands available for Windows VMs. You can use the `RunPowerShellScript` command to run any custom script that you want. When you're using the Azure CLI or PowerShell to run a command, the value that you provide for the `--command-id` or `-CommandId` parameter must be one of the following listed values. When you specify a value that is not an available command, you receive this error:

The entity was not found in this Azure location

NAME	DESCRIPTION
RunPowerShellScript	Runs a PowerShell script
DisableNLA	Disable Network Level Authentication
DisableWindowsUpdate	Disable Windows Update Automatic Updates
EnableAdminAccount	Checks if the local administrator account is disabled, and if so enables it.
EnableEMS	EnableS EMS
EnableRemotePS	Configures the machine to enable remote PowerShell.
EnableWindowsUpdate	Enable Windows Update Automatic Updates
IPConfig	Shows detailed information for the IP address, subnet mask, and default gateway for each adapter bound to TCP/IP.
RDPSetting	Checks registry settings and domain policy settings. Suggests policy actions if the machine is part of a domain or modifies the settings to default values.
ResetRDPCert	Removes the TLS/SSL certificate tied to the RDP listener and restores the RDP listener security to default. Use this script if you see any issues with the certificate.
SetRDPPort	Sets the default or user-specified port number for Remote Desktop connections. Enables firewall rules for inbound access to the port.

## Azure CLI

The following example uses the [az vm run-command](#) command to run a shell script on an Azure Windows VM.

```
# script.ps1
# param(
#     [string]$arg1,
#     [string]$arg2
# )
# Write-Host This is a sample script with parameters $arg1 and $arg2

az vm run-command invoke --command-id RunPowerShellScript --name win-vm -g my-resource-group \
--scripts @script.ps1 --parameters "arg1=somefoo" "arg2=somebar"
```

## Azure portal

Go to a VM in the [Azure portal](#) and select **Run command** from the left menu, under **Operations**. You see a list of the available commands to run on the VM.

 WindowsVM1 - Run command

Virtual machine

Search (Ctrl+ /)

- Disaster recovery
- Update management
- Inventory
- Change tracking
- Run command

Monitoring

- Alerts
- Metrics
- Diagnostics settings
- Advisor recommendations

Run Command uses the VM agent to let you run a script inside this virtual machine. This can be helpful for troubleshooting and recovery, and for general machine and application maintenance. Select a command below to see details.

NAME	DESCRIPTION
RunPowerShellScript	Executes a PowerShell script
EnableAdminAccount	Enable administrator account
EnableRemotePS	Enable remote PowerShell
IPConfig	List IP configuration
RDPSettings	Verify RDP Listener Settings
ResetRDPCert	Restore RDP Authentication mode to defaults
SetRDPPort	Set Remote Desktop port

Learn more  
Run Command  
Provide feedback

Choose a command to run. Some of the commands might have optional or required input parameters. For those commands, the parameters are presented as text fields for you to provide the input values. For each command, you can view the script that's being run by expanding **View script**. **RunPowerShellScript** is different from the other commands, because it allows you to provide your own custom script.

**NOTE**

The built-in commands are not editable.

After you choose the command, select **Run** to run the script. After the script finishes, it returns the output and any errors in the output window. The following screenshot shows an example output from running the **RDPSettings** command.

Run Command Script

RDPSettings

 Script execution complete

Details  
Checks registry settings and domain policy settings. Suggests policy actions if machine is part of a domain or modifies the settings to default values.

View script

Parameters  
No parameters

**Run**

Output

```
Not domain joined
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\PortNumber: 3389
HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services\fDenyTSConnections:
HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services\KeepAliveEnable: 1
HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services\KeepAliveInterval: 1
HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services\KeepAliveTimeout: 1
HKLM:\SOFTWARE\Policies\Microsoft\Windows NT\Terminal Services\fDisableAutoReconnect: 0
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\fInheritReconnectSame: 1
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\fReconnectSame: 0
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\fInheritMaxSessionTime: 1
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\fInheritMaxDisconnectionTime: 1
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\MaxDisconnectionTime: 0
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\MaxConnectionTime: 0
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\fInheritMaxIdleTime: 1
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\MaxIdleTime: 0
HKLM:\SYSTEM\CurrentControlSet\Control\Terminal Server\Winstations\RDP-Tcp\MaxInstanceCount: 4294967295
```

## PowerShell

The following example uses the [Invoke-AzVMRunCommand](#) cmdlet to run a PowerShell script on an Azure VM. The cmdlet expects the script referenced in the `-ScriptPath` parameter to be local to where the cmdlet is being run.

```
Invoke-AzVMRunCommand -ResourceGroupName '<myResourceGroup>' -Name '<myVMName>' -CommandId  
'RunPowerShellScript' -ScriptPath '<pathToScript>' -Parameter @{"arg1" = "var1";"arg2" = "var2"}
```

### NOTE

Parameter values can be string type only and the script is responsible for converting them to other types if needed.

## Limiting access to Run Command

Listing the run commands or showing the details of a command requires the

[Microsoft.Compute/locations/runCommands/read](#) permission on Subscription Level. The built-in [Reader](#) role and higher levels have this permission.

Running a command requires the [Microsoft.Compute/virtualMachines/runCommand/action](#) permission. The [Virtual Machine Contributor](#) role and higher levels have this permission.

You can use one of the [built-in roles](#) or create a [custom role](#) to use Run Command.

## Next steps

To learn about other ways to run scripts and commands remotely in your VM, see [Run scripts in your Windows VM](#).

# Preview: Run scripts in your Linux VM by using managed Run Commands

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

## IMPORTANT

Managed Run Command is currently in public preview. This preview version is provided without a service-level agreement, and we don't recommend it for production workloads. Certain features might not be supported or might have constrained capabilities. For more information, see [Supplemental Terms of Use for Microsoft Azure Previews](#).

The Run Command feature uses the virtual machine (VM) agent to run scripts within an Azure Linux VM. You can use these scripts for general machine or application management. They can help you quickly diagnose and remediate VM access and network issues and get the VM back to a good state.

The *updated* managed Run Command uses the same VM agent channel to execute scripts and provides the following enhancements over the [original action oriented Run Command](#):

- Support for updated Run Command through ARM deployment template
- Parallel execution of multiple scripts
- Sequential execution of scripts
- RunCommand script can be canceled
- User specified script timeout
- Support for long running (hours/days) scripts
- Passing secrets (parameters, passwords) in a secure manner

## Register for preview

You must register your subscription in order to use managed Run Command during public preview. Go to [set up preview features in Azure subscription](#) for registration instructions and use the feature name `RunCommandPreview`.

## Azure CLI

The following examples use `az vm run-command` to run shell script on an Azure Linux VM.

### Execute a script with the VM

This command will deliver the script to the VM, execute it, and return the captured output.

```
az vm run-command create --name "myRunCommand" --vm-name "myVM" --resource-group "myRG" --script "echo Hello World!"
```

### List all deployed RunCommand resources on a VM

This command will return a full list of previously deployed Run Commands along with their properties.

```
az vm run-command list --vm-name "myVM" --resource-group "myRG"
```

## Get execution status and results

This command will retrieve current execution progress, including latest output, start/end time, exit code, and terminal state of the execution.

```
az vm run-command show --name "myRunCommand" --vm-name "myVM" --resource-group "myRG" --expand instanceView
```

## Delete RunCommand resource from the VM

Remove the RunCommand resource previously deployed on the VM. If the script execution is still in progress, execution will be terminated.

```
az vm run-command delete --name "myRunCommand" --vm-name "myVM" --resource-group "myRG"
```

# PowerShell

## Execute a script with the VM

This command will deliver the script to the VM, execute it, and return the captured output.

```
Set-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -Location "EastUS" -RunCommandName "RunCommandName" -SourceScript "echo Hello World!"
```

## Execute a script on the VM using SourceScriptUri parameter

`OutputBlobUri` and `ErrorBlobUri` are optional parameters.

```
Set-AzVMRunCommand -ResourceGroupName -VMName -RunCommandName -SourceScriptUri "< SAS URI of a storage blob with read access or public URI>" -OutputBlobUri "< SAS URI of a storage append blob with read, add, create, write access>" -ErrorBlobUri "< SAS URI of a storage append blob with read, add, create, write access>"
```

## List all deployed RunCommand resources on a VM

This command will return a full list of previously deployed Run Commands along with their properties.

```
Get-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM"
```

## Get execution status and results

This command will retrieve current execution progress, including latest output, start/end time, exit code, and terminal state of the execution.

```
Get-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -RunCommandName "RunCommandName" -Expand instanceView
```

## Delete RunCommand resource from the VM

Remove the RunCommand resource previously deployed on the VM. If the script execution is still in progress, execution will be terminated.

```
Remove-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -RunCommandName "RunCommandName"
```

# REST API

To deploy a new Run Command, execute a PUT on the VM directly and specify a unique name for the Run

Command instance.

```
GET  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachines/<vmName>/runcommands?api-version=2019-12-01
```

```
{  
  "location": "<location>",  
  "properties": {  
    "source": {  
      "script": "Write-Host Hello World!",  
      "scriptUri": "<SAS URI of a storage blob with read access or public URI>",  
      "commandId": "<Id>"  
    },  
    "parameters": [  
      {  
        "name": "param1",  
        "value": "value1"  
      },  
      {  
        "name": "param2",  
        "value": "value2"  
      }  
    ],  
    "protectedParameters": [  
      {  
        "name": "secret1",  
        "value": "value1"  
      },  
      {  
        "name": "secret2",  
        "value": "value2"  
      }  
    ],  
    "runAsUser": "userName",  
    "runAsPassword": "userPassword",  
    "timeoutInSeconds": 3600,  
    "outputBlobUri": "< SAS URI of a storage append blob with read, add, create, write access>",  
    "errorBlobUri": "< SAS URI of a storage append blob with read, add, create, write access >"  
  }  
}
```

## Notes

- You can provide an inline script, a script URI, or a built-in script [command ID](#) as the input source. Script URI is either storage blob SAS URI with read access or public URI.
- Only one type of source input is supported for one command execution.
- Run Command supports writing output and error to Storage blobs using outputBlobUri and errorBlobUri parameters, which can be used to store large script outputs. Use SAS URI of a storage append blob with read, add, create, write access. The blob should be of type AppendBlob. Writing the script output or error blob would fail otherwise. The blob will be overwritten if it already exists. It will be created if it does not exist.

## List running instances of Run Command on a VM

```
GET  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachines/<vmName>/runcommands?api-version=2019-12-01
```

## Get output details for a specific Run Command deployment

```
GET  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands/<runCommandName>?$expand=instanceView&api-version=2019-12-01
```

## Cancel a specific Run Command deployment

To cancel a running deployment, you can PUT or PATCH on the running instance of Run Command and specify a blank script in the request body. This will cancel the ongoing execution.

You can also delete the instance of Run Command.

```
DELETE  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands/<runCommandName>?api-version=2019-12-01
```

## Deploy scripts in an ordered sequence

To deploy scripts sequentially, use a deployment template, specifying a `dependsOn` relationship between sequential scripts.

```
{  
    "type": "Microsoft.Compute/virtualMachines/runCommands",  
    "name": "secondRunCommand",  
    "apiVersion": "2019-12-01",  
    "location": "[parameters('location')]",  
    "dependsOn": <full resourceId of the previous other Run Command>,  
    "properties": {  
        "source": {  
            "script": "echoHelloWorld!"  
        },  
        "timeoutInSeconds": 60  
    }  
}
```

## Execute multiple Run Commands sequentially

By default, if you deploy multiple RunCommand resources using deployment template, they will be executed simultaneously on the VM. If you have a dependency on the scripts and a preferred order of execution, you can use the `dependsOn` property to make them run sequentially.

In this example, `secondRunCommand` will execute after `firstRunCommand`.

```
{
    "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "resources": [
        {
            "type": "Microsoft.Compute/virtualMachines/runCommands",
            "name": "[concat(parameters('vmName'), '/firstRunCommand')]",
            "apiVersion": "2019-12-01",
            "location": "[parameters('location')]",
            "dependsOn": [
                "[concat('Microsoft.Compute/virtualMachines/', parameters('vmName'))]"
            ],
            "properties": {
                "source": {
                    "script": "echo First: Hello World!"
                },
                "parameters": [
                    {
                        "name": "param1",
                        "value": "value1"
                    },
                    {
                        "name": "param2",
                        "value": "value2"
                    }
                ],
                "timeoutInSeconds": 20
            }
        },
        {
            "type": "Microsoft.Compute/virtualMachines/runCommands",
            "name": "[concat(parameters('vmName'), '/secondRunCommand')]",
            "apiVersion": "2019-12-01",
            "location": "[parameters('location')]",
            "dependsOn": [
                "[concat('Microsoft.Compute/virtualMachines/',
parameters('vmName'), 'runcommands/firstRunCommand')]"
            ],
            "properties": {
                "source": {
                    "scriptUri": "http://github.com/myscript.ps1"
                },
                "timeoutInSeconds": 60
            }
        }
    ]
}
```

## Next steps

To learn about other ways to run scripts and commands remotely in your VM, see [Run scripts in your Linux VM](#).

# Preview: Run scripts in your Windows VM by using managed Run Commands

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

## IMPORTANT

Managed Run Command is currently in public preview. This preview version is provided without a service-level agreement, and we don't recommend it for production workloads. Certain features might not be supported or might have constrained capabilities. For more information, see [Supplemental Terms of Use for Microsoft Azure Previews](#).

The Run Command feature uses the virtual machine (VM) agent to scripts within an Azure Windows VM. You can use these scripts for general machine or application management. They can help you quickly diagnose and remediate VM access and network issues and get the VM back to a good state.

The *updated* managed Run Command uses the same VM agent channel to execute scripts and provides the following enhancements over the [original action orientated Run Command](#):

- Support for updated Run Command through ARM deployment template
- Parallel execution of multiple scripts
- Sequential execution of scripts
- RunCommand script can be canceled
- User specified script timeout
- Support for long running (hours/days) scripts
- Passing secrets (parameters, passwords) in a secure manner

## Register for preview

You must register your subscription in order to use managed Run Command during public preview. Go to [set up preview features in Azure subscription](#) for registration instructions and use the feature name `RunCommandPreview`

## Azure CLI

The following examples use `az vm run-command` to run shell script on an Azure Windows VM.

### Execute a script with the VM

This command will deliver the script to the VM, execute it, and return the captured output.

```
az vm run-command create --name "myRunCommand" --vm-name "myVM" --resource-group "myRG" --script "Write-Host Hello World!"
```

### List all deployed RunCommand resources on a VM

This command will return a full list of previously deployed Run Commands along with their properties.

```
az vm run-command list --vm-name "myVM" --resource-group "myRG"
```

## Get execution status and results

This command will retrieve current execution progress, including latest output, start/end time, exit code, and terminal state of the execution.

```
az vm run-command show --name "myRunCommand" --vm-name "myVM" --resource-group "myRG" --expand instanceView
```

## Delete RunCommand resource from the VM

Remove the RunCommand resource previously deployed on the VM. If the script execution is still in progress, execution will be terminated.

```
az vm run-command delete --name "myRunCommand" --vm-name "myVM" --resource-group "myRG"
```

# PowerShell

## Execute a script with the VM

This command will deliver the script to the VM, execute it, and return the captured output.

```
Set-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -Location "EastUS" -RunCommandName "RunCommandName" -SourceScript "echo Hello World!"
```

## Execute a script on the VM using SourceScriptUri parameter

`OutputBlobUri` and `ErrorBlobUri` are optional parameters.

```
Set-AzVMRunCommand -ResourceGroupName -VMName -RunCommandName -SourceScriptUri "< SAS URI of a storage blob with read access or public URI>" -OutputBlobUri "< SAS URI of a storage append blob with read, add, create, write access>" -ErrorBlobUri "< SAS URI of a storage append blob with read, add, create, write access>"
```

## List all deployed RunCommand resources on a VM

This command will return a full list of previously deployed Run Commands along with their properties.

```
Get-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM"
```

## Get execution status and results

This command will retrieve current execution progress, including latest output, start/end time, exit code, and terminal state of the execution.

```
Get-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -RunCommandName "RunCommandName" -Status
```

## Delete RunCommand resource from the VM

Remove the RunCommand resource previously deployed on the VM. If the script execution is still in progress, execution will be terminated.

```
Remove-AzVMRunCommand -ResourceGroupName "myRG" -VMName "myVM" -RunCommandName "RunCommandName"
```

# REST API

To deploy a new Run Command, execute a PUT on the VM directly and specify a unique name for the Run

Command instance.

```
PUT  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands/<runCommandName>?api-version=2019-12-01
```

```
{  
    "location": "<location>",  
    "properties": {  
        "source": {  
            "script": "Write-Host Hello World!",  
            "scriptUri": "<SAS URI of a storage blob with read access or public URI>",  
            "commandId": "<Id>"  
        },  
        "parameters": [  
            {  
                "name": "param1",  
                "value": "value1"  
            },  
            {  
                "name": "param2",  
                "value": "value2"  
            }  
        ],  
        "protectedParameters": [  
            {  
                "name": "secret1",  
                "value": "value1"  
            },  
            {  
                "name": "secret2",  
                "value": "value2"  
            }  
        ],  
        "runAsUser": "userName",  
        "runAsPassword": "userPassword",  
        "timeoutInSeconds": 3600,  
        "outputBlobUri": "< SAS URI of a storage append blob with read, add, create, write access>",  
        "errorBlobUri": "< SAS URI of a storage append blob with read, add, create, write access >"  
    }  
}
```

## Notes

- You can provide an inline script, a script URI, or a built-in script [command ID](#) as the input source. Script URI is either storage blob SAS URI with read access or public URI.
- Only one type of source input is supported for one command execution.
- Run Command supports writing output and error to Storage blobs using outputBlobUri and errorBlobUri parameters, which can be used to store large script outputs. Use SAS URI of a storage append blob with read, add, create, write access. The blob should be of type AppendBlob. Writing the script output or error blob would fail otherwise. The blob will be overwritten if it already exists. It will be created if it does not exist.

## List running instances of Run Command on a VM

```
GET  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands?api-version=2019-12-01
```

## Get output details for a specific Run Command deployment

```
GET  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands/<runCommandName>?$expand=instanceView&api-version=2019-12-01
```

## Cancel a specific Run Command deployment

To cancel a running deployment, you can PUT or PATCH on the running instance of Run Command and specify a blank script in the request body. This will cancel the ongoing execution.

You can also delete the instance of Run Command.

```
DELETE  
/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachin  
es/<vmName>/runcommands/<runCommandName>?api-version=2019-12-01
```

## Deploy scripts in an ordered sequence

To deploy scripts sequentially, use a deployment template, specifying a `dependsOn` relationship between sequential scripts.

```
{  
    "type": "Microsoft.Compute/virtualMachines/runCommands",  
    "name": "secondRunCommand",  
    "apiVersion": "2019-12-01",  
    "location": "[parameters('location')]",  
    "dependsOn": <full resourceId of the previous other Run Command>,  
    "properties": {  
        "source": {  
            "script": "Write-HostHelloWorld!"  
        },  
        "timeoutInSeconds": 60  
    }  
}
```

## Execute multiple Run Commands sequentially

By default, if you deploy multiple RunCommand resources using deployment template, they will be executed simultaneously on the VM. If you have a dependency on the scripts and a preferred order of execution, you can use the `dependsOn` property to make them run sequentially.

In this example, `secondRunCommand` will execute after `firstRunCommand`.

```
{
    "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "resources": [
        {
            "type": "Microsoft.Compute/virtualMachines/runCommands",
            "name": "[concat(parameters('vmName'), '/firstRunCommand')]",
            "apiVersion": "2019-12-01",
            "location": "[parameters('location')]",
            "dependsOn": [
                "[concat('Microsoft.Compute/virtualMachines/', parameters('vmName'))]"
            ],
            "properties": {
                "source": {
                    "script": "Write-Host First: Hello World!"
                },
                "parameters": [
                    {
                        "name": "param1",
                        "value": "value1"
                    },
                    {
                        "name": "param2",
                        "value": "value2"
                    }
                ],
                "timeoutInSeconds": 20
            }
        },
        {
            "type": "Microsoft.Compute/virtualMachines/runCommands",
            "name": "[concat(parameters('vmName'), '/secondRunCommand')]",
            "apiVersion": "2019-12-01",
            "location": "[parameters('location')]",
            "dependsOn": [
                "[concat('Microsoft.Compute/virtualMachines/',
parameters('vmName'), 'runcommands/firstRunCommand')]"
            ],
            "properties": {
                "source": {
                    "scriptUri": "http://github.com/myscript.ps1"
                },
                "timeoutInSeconds": 60
            }
        }
    ]
}
```

## Next steps

To learn about other ways to run scripts and commands remotely in your VM, see [Run scripts in your Windows VM](#).

# Azure virtual machine extensions and features

9/21/2022 • 2 minutes to read • [Edit Online](#)

Extensions are small applications that provide post-deployment configuration and automation on Azure VMs. The Azure platform hosts many extensions covering VM configuration, monitoring, security, and utility applications. Publishers take an application, wrap it into an extension, and simplify the installation. All you need to do is provide mandatory parameters.

## How can I find what extensions are available?

You can view available extensions by selecting a VM, the selecting **Extensions** in the left menu. To pull a full list of extensions, see [Discovering VM Extensions for Linux](#) and [Discovering VM Extensions for Windows](#).

## How can I install an extension?

Azure VM extensions can be managed using the Azure CLI, PowerShell, Resource Manager templates, and the Azure portal. To try an extension, go to the Azure portal, select the Custom Script Extension, then pass in a command or script to run the extension.

For more information, see [Windows Custom Script Extension](#) and [Linux Custom Script Extension](#).

## How do I manage extension application lifecycle?

You do not need to connect to a VM directly to install or delete an extension. The Azure extension lifecycle is managed outside of the VM and integrated into the Azure platform.

## Anything else I should be thinking about for extensions?

Some individual VM extension applications may have their own environmental prerequisites, such as access to an endpoint. Each extension has an article that explains any pre-requisites, including which operating systems are supported.

## Troubleshoot extensions

If you are looking for general troubleshooting steps for Windows VM extensions, please refer to [Troubleshooting Azure Windows VM extension failures](#).

Otherwise, specific troubleshooting information for each extension can be found in the **Troubleshoot and support** section in the overview for the extension. Here is a list of the troubleshooting information available:

NAMESPACE	TROUBLESHOOTING
microsoft.azure.monitoring.dependencyagent.dependencyagentlinux	<a href="#">Azure Monitor Dependency for Linux</a>
microsoft.azure.monitoring.dependencyagent.dependencyagentwindows	<a href="#">Azure Monitor Dependency for Windows</a>
microsoft.azure.security.azurediskencryptionforlinux	<a href="#">Azure Disk Encryption for Linux</a>

NAMESPACE	TROUBLESHOOTING
microsoft.azure.security.azurediskencryption	<a href="#">Azure Disk Encryption for Windows</a>
microsoft.compute.customscriptextension	<a href="#">Custom Script for Windows</a>
microsoft.ostcextensions.customscriptforlinux	<a href="#">Desired State Configuration for Linux</a>
microsoft.powershell.dsc	<a href="#">Desired State Configuration for Windows</a>
microsoft.hpccompute.nvidiagpudriverlinux	<a href="#">NVIDIA GPU Driver Extension for Linux</a>
microsoft.hpccompute.nvidiagpudriverwindows	<a href="#">NVIDIA GPU Driver Extension for Windows</a>
microsoft.azure.security.iaasantimalware	<a href="#">Antimalware Extension for Windows</a>
microsoft.enterprisecloud.monitoring.omsagentforlinux	<a href="#">Azure Monitor for Linux</a>
microsoft.enterprisecloud.monitoring.microsoftmonitoringagent	<a href="#">Azure Monitor for Windows</a>
stackify.linuxagent.extension.stackifylinuxagentextension	<a href="#">Stackify Retrace for Linux</a>
vmaccessforlinux.microsoft.ostcextensions	<a href="#">Reset password for Linux</a>
microsoft.recoveryservices.vmsnapshot	<a href="#">Snapshot for Linux</a>
microsoft.recoveryservices.vmsnapshot	<a href="#">Snapshot for Windows</a>

## Next steps

- For more information about how the Linux Agent and extensions work, see [Azure VM extensions and features for Linux](#).
- For more information about how the Windows Guest Agent and extensions work, see [Azure VM extensions and features for Windows](#).
- To install the Windows Guest Agent, see [Azure Windows Virtual Machine Agent Overview](#).
- To install the Linux Agent, see [Azure Linux Virtual Machine Agent Overview](#).

# Virtual machine extensions and features for Linux

9/21/2022 • 13 minutes to read • [Edit Online](#)

Azure virtual machine (VM) extensions are small applications that provide post-deployment configuration and automation tasks on Azure VMs. For example, if a virtual machine requires software installation, antivirus protection, or the ability to run a script inside it, you can use a VM extension.

You can run Azure VM extensions by using the Azure CLI, PowerShell, Azure Resource Manager templates (ARM templates), and the Azure portal. You can bundle extensions with a new VM deployment or run them against any existing system.

This article provides an overview of Azure VM extensions, prerequisites for using them, and guidance on how to detect, manage, and remove them. This article provides generalized information because many VM extensions are available. Each has a potentially unique configuration and its own documentation.

## Use cases and samples

Each Azure VM extension has a specific use case. Examples include:

- Apply PowerShell desired state configurations (DSCs) to a VM by using the [DSC extension for Linux](#).
- Configure monitoring of a VM by using the [Microsoft Monitoring Agent VM extension](#).
- Configure monitoring of your Azure infrastructure by using the [Chef](#) or [Datadog](#) extension.

In addition to process-specific extensions, a Custom Script extension is available for both Windows and Linux virtual machines. The [Custom Script extension for Linux](#) allows any Bash script to be run on a VM. Custom scripts are useful for designing Azure deployments that require configuration beyond what native Azure tooling can provide.

## Prerequisites

### Azure Linux Agent

To handle the extension on the VM, you need the [Azure Linux Agent](#) installed. Some individual extensions have prerequisites, such as access to resources or dependencies.

The Azure Linux Agent manages interactions between an Azure VM and the Azure fabric controller. The agent is responsible for many functional aspects of deploying and managing Azure VMs, including running VM extensions.

The Azure Linux Agent is preinstalled on Azure Marketplace images. It can also be installed manually on supported operating systems.

The agent runs on multiple operating systems. However, the extensions framework has a [limit for the operating systems that extensions use](#). Some extensions are not supported across all operating systems and might emit error code 51 ("Unsupported OS"). Check the individual extension documentation for supportability.

### Network access

Extension packages are downloaded from the Azure Storage extension repository. Extension status uploads are posted to Azure Storage.

If you use a [supported version of the Azure Linux Agent](#), you don't need to allow access to Azure Storage in the VM region. You can use the agent to redirect the communication to the Azure fabric controller for agent communications. If you're on an unsupported version of the agent, you need to allow outbound access to Azure

Storage in that region from the VM.

#### IMPORTANT

If you've blocked access to the private IP address 168.63.129.16 by using the guest firewall, extensions fail even if you're using a supported version of the agent or you've configured outbound access.

Agents can only be used to download extension packages and reporting status. For example, if an extension installation needs to download a script from GitHub (Custom Script extension) or needs access to Azure Storage (Azure Backup), then you need to open additional firewall or network security group (NSG) ports. Different extensions have different requirements, because they're applications in their own right. For extensions that require access to Azure Storage, you can allow access by using Azure NSG [service tags](#).

To redirect agent traffic requests, the Azure Linux Agent has proxy server support. However, this proxy server support does not apply extensions. You must configure each individual extension to work with a proxy.

## Discover VM extensions

- [Azure CLI](#)
- [Azure PowerShell](#)

Many VM extensions are available for use with Azure VMs. To see a complete list, use [az vm extension image list](#). The following example lists all available extensions in the *westus* location:

```
az vm extension image list --location westus --output table
```

## Run VM extensions

Azure VM extensions run on existing VMs. That's useful when you need to make configuration changes or recover connectivity on an already deployed VM. VM extensions can also be bundled with ARM template deployments. By using extensions with ARM templates, you can deploy and configure Azure VMs without post-deployment intervention.

You can use the following methods to run an extension against an existing VM.

### Azure CLI

You can run Azure VM extensions against an existing VM by using the [az vm extension set](#) command. The following example runs the Custom Script extension against a VM named *myVM* in a resource group named *myResourceGroup*. Replace the example resource group name, VM name, and script to run (<https://raw.githubusercontent.com/me/project/hello.sh>) with your own information.

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name customScript \
--publisher Microsoft.Azure.Extensions \
--settings '{"fileUris": ["https://raw.githubusercontent.com/me/project/hello.sh"], "commandToExecute": \
"./hello.sh"}'
```

When the extension runs correctly, the output is similar to the following example:

```
info: Executing command vm extension set
+ Looking up the VM "myVM"
+ Installing extension "CustomScript", VM: "myVM"
info: vm extension set command OK
```

## Azure PowerShell

You can run Azure VM extensions against an existing VM by using the [Set-AzVMExtension](#) command. The following example runs the Custom Script extension against a VM named *myVM* in a resource group named *myResourceGroup*. Replace the example resource group name, VM name, and script to run (<https://raw.githubusercontent.com/me/project/hello.sh>) with your own information.

```
$Params = @{
    ResourceGroupName = 'myResourceGroup'
    VMName           = 'myVM'
    Name              = 'CustomScript'
    Publisher         = 'Microsoft.Azure.Extensions'
    ExtensionType     = 'CustomScript'
    TypeHandlerVersion = '2.1'
    Settings          = @{'fileUris' = @('https://raw.githubusercontent.com/me/project/hello.sh');
    commandToExecute = './hello.sh'}
}
Set-AzVMExtension @Params
```

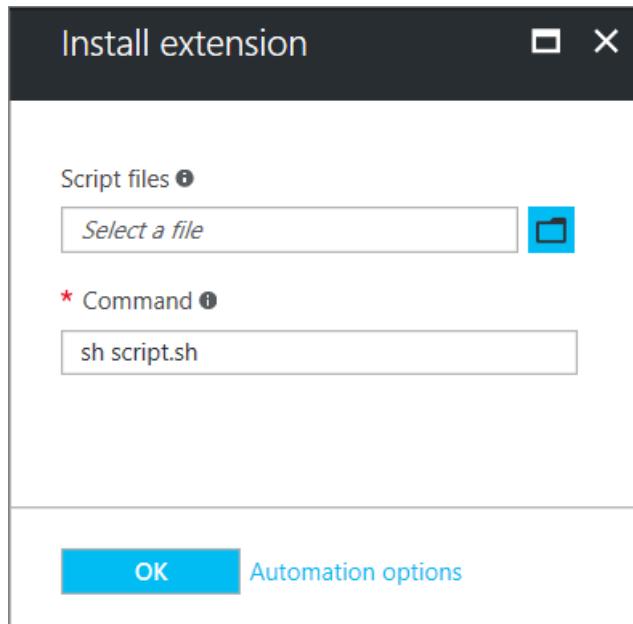
When the extension runs correctly, the output is similar to the following example:

RequestId	IsSuccess	StatusCode	StatusCode	ReasonPhrase
	True	OK	OK	

## Azure portal

You can apply VM extensions to an existing VM through the Azure portal. Select the VM in the portal, select **Extensions**, and then select **Add**. Choose the extension that you want from the list of available extensions, and follow the instructions in the wizard.

The following image shows the installation of the Custom Script extension for Linux from the Azure portal:



## Azure Resource Manager templates

You can add VM extensions to an ARM template and run them with the deployment of the template. When you deploy an extension with a template, you can create fully configured Azure deployments.

For example, the following JSON is taken from a [full ARM template](#) that deploys a set of load-balanced VMs and an Azure SQL database, and then installs a .NET Core application on each VM. The VM extension takes care of the software installation.

```
{  
    "apiVersion": "2015-06-15",  
    "type": "extensions",  
    "name": "config-app",  
    "location": "[resourceGroup().location]",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', concat(variables('vmName'),copyindex()))]"  
    ],  
    "tags": {  
        "displayName": "config-app"  
    },  
    "properties": {  
        "publisher": "Microsoft.Azure.Extensions",  
        "type": "CustomScript",  
        "typeHandlerVersion": "2.1",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "fileUris": [  
                "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-  
linux/scripts/config-music.sh"  
            ]  
        },  
        "protectedSettings": {  
            "commandToExecute": "[concat('sudo sh config-music.sh ',variables('musicStoreSqlName'), ' ',  
parameters('adminUsername'), ' ', parameters('sqlAdminPassword'))]"  
        }  
    }  
}
```

For more information on creating ARM templates, see [Virtual machines in an Azure Resource Manager template](#).

## Help secure VM extension data

When you run a VM extension, it might be necessary to include sensitive information such as credentials, storage account names, and access keys. Many VM extensions include a protected configuration that encrypts data and only decrypts it inside the target VM. Each extension has a specific protected configuration schema, and each is detailed in extension-specific documentation.

The following example shows an instance of the Custom Script extension for Linux. The command to run includes a set of credentials. In this example, the command to run is not encrypted.

```
{
  "apiVersion": "2015-06-15",
  "type": "extensions",
  "name": "config-app",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', concat(variables('vmName'),copyindex()))]"
  ],
  "tags": {
    "displayName": "config-app"
  },
  "properties": {
    "publisher": "Microsoft.Azure.Extensions",
    "type": "CustomScript",
    "typeHandlerVersion": "2.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "fileUris": [
        "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-
linux/scripts/config-music.sh"
      ],
      "commandToExecute": "[concat('sudo sh config-music.sh ',variables('musicStoreSqlName'), ' ',parameters('adminUsername'), ' ', parameters('sqlAdminPassword'))]"
    }
  }
}
```

Moving the `commandToExecute` property to the `protected` configuration helps secure the execution string, as shown in the following example:

```
{
  "apiVersion": "2015-06-15",
  "type": "extensions",
  "name": "config-app",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', concat(variables('vmName'),copyindex()))]"
  ],
  "tags": {
    "displayName": "config-app"
  },
  "properties": {
    "publisher": "Microsoft.Azure.Extensions",
    "type": "CustomScript",
    "typeHandlerVersion": "2.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "fileUris": [
        "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-
linux/scripts/config-music.sh"
      ]
    },
    "protectedSettings": {
      "commandToExecute": "[concat('sudo sh config-music.sh ',variables('musicStoreSqlName'), ' ',parameters('adminUsername'), ' ', parameters('sqlAdminPassword'))]"
    }
  }
}
```

## How agents and extensions are updated

Agents and extensions share the same automatic update mechanism.

When an update is available and automatic updates are enabled, the update is installed on the VM only after

there's a change to an extension or after other VM model changes, such as:

- Data disks
- Extensions
- Extension Tags
- Boot diagnostics container
- Guest OS secrets
- VM size
- Network profile

Publishers make updates available to regions at various times, so it's possible that you can have VMs in different regions on different versions.

#### NOTE

Some updates might require additional firewall rules. See [Network access](#).

#### Agent updates

The Linux VM Agent contains *Provisioning Agent code* and *extension-handling code* in one package. They can't be separated.

You can disable the Provisioning Agent when you want to [provision on Azure by using cloud-init](#).

Supported versions of the agents can use automatic updates. The only code that can be updated is the extension-handling code, not the Provisioning Agent code. The Provisioning Agent code is run-once code.

The extension-handling code is responsible for:

- Communicating with the Azure fabric.
- Handling the VM extension operations, such as installations, reporting status, updating the individual extensions, and removing extensions. Updates contain security fixes, bug fixes, and enhancements to the extension-handling code.

When the agent is installed, a parent daemon is created. This parent then spawns a child process that's used to handle extensions. If an update is available for the agent, it's downloaded. The parent stops the child process, upgrades it, and then restarts it. If there's a problem with the update, the parent process rolls back to the previous child version.

The parent process can't be automatically updated. The parent can be updated only by a distribution package update.

To check what version you're running, check `waagent` as follows:

```
waagent --version
```

The output is similar to the following example:

```
WALinuxAgent-2.2.45 running on ubuntu 18.04
Python: 3.6.9
Goal state agent: 2.7.1.0
```

In the preceding example output, the parent (or package deployed version) is `WALinuxAgent-2.2.45`. The `Goal state agent` value is the auto-update version.

We highly recommend that you always enable automatic update for the agent: `AutoUpdate.Enabled=y`. If you

don't enable automatic update, you'll need to keep manually updating the agent, and you won't get bug and security fixes.

#### Extension updates

When an extension update is available and automatic updates are enabled, after a [change to the VM model](#) occurs, the Azure Linux Agent downloads and upgrades the extension.

Automatic extension updates are either *minor* or *hotfix*. You can opt in or opt out of minor updates when you provision the extension. The following example shows how to automatically upgrade minor versions in an ARM template by using `"autoUpgradeMinorVersion": true,`:

```
"publisher": "Microsoft.Azure.Extensions",
"type": "CustomScript",
"typeHandlerVersion": "2.1",
"autoUpgradeMinorVersion": true,
"settings": {
    "fileUris": [
        "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-
linux/scripts/config-music.sh"
    ]
},
```

To get the latest minor-release bug fixes, we highly recommend that you always select automatic update in your extension deployments. You can't opt out of hotfix updates that carry security or key bug fixes.

If you disable automatic updates or you need to upgrade a major version, use [az vm extension set](#) or [Set-AzVMExtension](#) and specify the target version.

#### How to identify extension updates

##### Identify if the extension is set with autoUpgradeMinorVersion on a VM

- [Azure CLI](#)
- [Azure PowerShell](#)

You can see from the VM model if the extension was provisioned with `autoUpgradeMinorVersion`. To check, use [az vm show](#) and provide the resource group and VM name as follows:

```
az vm show --resource-group myResourceGroup --name myVM
```

The following example output shows that `autoUpgradeMinorVersion` is set to `true`:

```
"resources": [
    {
        "autoUpgradeMinorVersion": true,
        "forceUpdateTag": null,
        "id": "/subscriptions/guid/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM/extensi-
ons/customScript",
```

##### Identify when an autoUpgradeMinorVersion event occurred

To see when an update to the extension occurred, review the agent logs on the VM at `/var/log/waagent.log`.

In the following example, the VM had `Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9025` installed. A hotfix was available to `Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027`.

```
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Expected handler state: enabled
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Decide which version to use
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Use version: 2.3.9027
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Current handler state is: NotInstalled
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Download extension package
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Unpack extension package
INFO Event: name=Microsoft.OSTCExtensions.LinuxDiagnostic, op=Download, message=Download succeeded
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Initialize extension directory
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Update settings file: 0.settings
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9025] Disable extension.
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9025] Launch command:diagnostic.py -disable
...
INFO Event: name=Microsoft.OSTCExtensions.LinuxDiagnostic, op=Disable, message=Launch command succeeded:
diagnostic.py -disable
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Update extension.
INFO [Microsoft.OSTCExtensions.LinuxDiagnostic-2.3.9027] Launch command:diagnostic.py -update
2017/08/14 20:21:57 LinuxAzureDiagnostic started to handle.
```

## Agent permissions

To perform its tasks, the agent needs to run as *root*.

## Troubleshoot VM extensions

Each VM extension might have specific troubleshooting steps. For example, when you use the Custom Script extension, you can find script execution details locally on the VM where the extension was run.

The following troubleshooting actions apply to all VM extensions:

- To check the Azure Linux Agent log, look at the activity when your extension was being provisioned in */var/log/waagent.log*.
- Check the extension logs for more details in */var/log/azure/<extensionName>*.
- Check troubleshooting sections in extension-specific documentation for error codes, known issues, and other extension-specific information.
- Look at the system logs. Check for other operations that might have interfered with the extension, such as a long-running installation of another application that required exclusive access to the package manager.

### Common reasons for extension failures

- Extensions have 20 minutes to run. (Exceptions are Custom Script, Chef, and DSC, which have 90 minutes.) If your deployment exceeds this time, it's marked as a timeout. The cause of this can be low-resource VMs, or other VM configurations or startup tasks are consuming large amounts of resources while the extension is trying to provision.
- Minimum prerequisites aren't met. Some extensions have dependencies on VM SKUs, such as HPC images. Extensions might have certain networking access requirements, such as communicating with Azure Storage or public services. Other examples might be access to package repositories, running out of disk space, or security restrictions.
- Package manager access is exclusive. In some cases, a long-running VM configuration and extension installation might conflict because they both need exclusive access to the package manager.

### View extension status

- [Azure CLI](#)
- [Azure PowerShell](#)

After a VM extension has been run against a VM, use [az vm get-instance-view](#) to return extension status as follows:

```
az vm get-instance-view \
--resource-group myResourceGroup \
--name myVM \
--query "instanceView.extensions"
```

The output is similar to the following example:

```
{
  "name": "customScript",
  "statuses": [
    {
      "code": "ProvisioningState/failed/0",
      "displayStatus": "Provisioning failed",
      "level": "Error",
      "message": "Enable failed: failed to execute command: command terminated with exit
status=127\n[stdout]\n\n[stderr]\n/bin/sh: 1: ech: not found\n",
      "time": null
    }
  ],
  "substatuses": null,
  "type": "Microsoft.Azure.Extensions.CustomScript",
  "typeHandlerVersion": "2.1.6"
}
```

You can also find extension execution status in the Azure portal. Select the VM, select **Extensions**, and then select the desired extension.

### Rerun a VM extension

There might be cases in which a VM extension needs to be rerun. You can rerun an extension by removing it, and then rerunning the extension with an execution method of your choice.

- [Azure CLI](#)
- [Azure PowerShell](#)

To remove an extension, use [az vm extension delete](#) as follows:

```
az vm extension delete \
--resource-group myResourceGroup \
--vm-name myVM \
--name customScript
```

You can also remove an extension in the Azure portal:

1. Select a VM.
2. Select **Extensions**.
3. Select the extension.
4. Select **Uninstall**.

## Common VM extension reference

EXTENSION NAME	DESCRIPTION
<a href="#">Custom Script extension for Linux</a>	Run scripts against an Azure virtual machine.

EXTENSION NAME	DESCRIPTION
<a href="#">VMAccess extension</a>	Regain access to an Azure virtual machine. You can also use it to <a href="#">manage users and credentials</a> .
<a href="#">Azure Diagnostics extension</a>	Manage Azure Diagnostics.

## Next steps

For more information about VM extensions, see [Azure virtual machine extensions and features](#).

# Understanding and using the Azure Linux Agent

9/21/2022 • 8 minutes to read • [Edit Online](#)

The Microsoft Azure Linux Agent (waagent) manages Linux & FreeBSD provisioning, and VM interaction with the Azure Fabric Controller. In addition to the Linux Agent providing provisioning functionality, Azure also provides the option of using cloud-init for some Linux OSes. The Linux Agent provides the following functionality for Linux and FreeBSD IaaS deployments:

## NOTE

For more information, see the [README](#).

- **Image Provisioning**

- Creation of a user account
- Configuring SSH authentication types
- Deployment of SSH public keys and key pairs
- Setting the host name
- Publishing the host name to the platform DNS
- Reporting SSH host key fingerprint to the platform
- Resource Disk Management
- Formatting and mounting the resource disk
- Configuring swap space

- **Networking**

- Manages routes to improve compatibility with platform DHCP servers
- Ensures the stability of the network interface name

- **Kernel**

- Configures virtual NUMA (disable for kernel < 2.6.37)
- Consumes Hyper-V entropy for /dev/random
- Configures SCSI timeouts for the root device (which could be remote)

- **Diagnostics**

- Console redirection to the serial port

- **SCVMM Deployments**

- Detects and bootstraps the VMM agent for Linux when running in a System Center Virtual Machine Manager 2012 R2 environment

- **VM Extension**

- Inject component authored by Microsoft and Partners into Linux VM (IaaS) to enable software and configuration automation
- VM Extension reference implementation on <https://github.com/Azure/azure-linux-extensions>

## Communication

The information flow from the platform to the agent occurs via two channels:

- A boot-time attached DVD for IaaS deployments. This DVD includes an OVF-compliant configuration file that includes all provisioning information other than the actual SSH keypairs.
- A TCP endpoint exposing a REST API used to obtain deployment and topology configuration.

## Requirements

The following systems have been tested and are known to work with the Azure Linux Agent:

### NOTE

This list may differ from the official list of [supported distros](#).

- CoreOS
- CentOS 6.3+
- Red Hat Enterprise Linux 6.7+
- Debian 7.0+
- Ubuntu 12.04+
- openSUSE 12.3+
- SLES 11 SP3+
- Oracle Linux 6.4+

Other Supported Systems:

- FreeBSD 10+ (Azure Linux Agent v2.0.10+)

The Linux agent depends on some system packages in order to function properly:

- Python 2.6+
- OpenSSL 1.0+
- OpenSSH 5.3+
- Filesystem utilities: sfdisk, fdisk, mkfs, parted
- Password tools: chpasswd, sudo
- Text processing tools: sed, grep
- Network tools: ip-route
- Kernel support for mounting UDF filesystems.

Ensure your VM has access to IP address 168.63.129.16. For more information, see [What is IP address 168.63.129.16](#).

## Installation

Installation using an RPM or a DEB package from your distribution's package repository is the preferred method of installing and upgrading the Azure Linux Agent. All the [endorsed distribution providers](#) integrate the Azure Linux agent package into their images and repositories.

Refer to the documentation in the [Azure Linux Agent repo on GitHub](#) for advanced installation options, such as installing from source or to custom locations or prefixes.

## Command-Line Options

### Flags

- verbose: Increase verbosity of specified command
- force: Skip interactive confirmation for some commands

## Commands

- help: Lists the supported commands and flags.
- deprovision: Attempt to clean the system and make it suitable for reprovisioning. The following operation deletes:
  - All SSH host keys (if Provisioning.RegenerateSshHostKeyPair is 'y' in the configuration file)
  - Nameserver configuration in `/etc/resolv.conf`
  - Root password from `/etc/shadow` (if Provisioning.DeleteRootPassword is 'y' in the configuration file)
  - Cached DHCP client leases
  - Resets host name to localhost.localdomain

### WARNING

Deprovisioning does not guarantee that the image is cleared of all sensitive information and suitable for redistribution.

- deprovision+user: Performs everything in -deprovision (above) and also deletes the last provisioned user account (obtained from `/var/lib/waagent`) and associated data. This parameter is when de-provisioning an image that was previously provisioning on Azure so it may be captured and reused.
- version: Displays the version of waagent
- serialconsole: Configures GRUB to mark ttyS0 (the first serial port) as the boot console. This ensures that kernel bootup logs are sent to the serial port and made available for debugging.
- daemon: Run waagent as a daemon to manage interaction with the platform. This argument is specified to waagent in the waagent init script.
- start: Run waagent as a background process

## Configuration

A configuration file (`/etc/waagent.conf`) controls the actions of waagent. The following shows a sample configuration file:

```
Provisioning.Enabled=y
Provisioning.DeleteRootPassword=n
Provisioning.RegenerateSshHostKeyPair=y
Provisioning.SshHostKeyPairType=rsa
Provisioning.MonitorHostName=y
Provisioning.DecodeCustomData=n
Provisioning.ExecuteCustomData=n
Provisioning.AllowResetSysUser=n
Provisioning.PasswordCryptId=6
Provisioning.PasswordCryptSaltLength=10
ResourceDisk.Format=y
ResourceDisk.Filesystem=ext4
ResourceDisk.MountPoint=/mnt/resource
ResourceDisk.MountOptions=None
ResourceDisk.EnableSwap=n
ResourceDisk.SwapSizeMB=0
LBProbeResponder=y
Logs.Verbose=n
OS.RootDeviceScsiTimeout=300
OS.OpensslPath=None
HttpProxy.Host=None
HttpProxy.Port=None
AutoUpdate.Enabled=y
```

The following various configuration options are described. Configuration options are of three types; Boolean,

String, or Integer. The Boolean configuration options can be specified as "y" or "n". The special keyword "None" may be used for some string type configuration entries as the following details:

#### Provisioning.Enabled:

Type: Boolean  
Default: y

This allows the user to enable or disable the provisioning functionality in the agent. Valid values are "y" or "n". If provisioning is disabled, SSH host and user keys in the image are preserved and any configuration specified in the Azure provisioning API is ignored.

#### NOTE

The `Provisioning.Enabled` parameter defaults to "n" on Ubuntu Cloud Images that use cloud-init for provisioning.

#### Provisioning.DeleteRootPassword:

Type: Boolean  
Default: n

If set, the root password in the /etc/shadow file is erased during the provisioning process.

#### Provisioning.RegenerateSshHostKeyPair:

Type: Boolean  
Default: y

If set, all SSH host key pairs (ecdsa, dsa, and rsa) are deleted during the provisioning process from `/etc/ssh/`. And a single fresh key pair is generated.

The encryption type for the fresh key pair is configurable by the `Provisioning.SshHostKeyPairType` entry. Some distributions re-create SSH key pairs for any missing encryption types when the SSH daemon is restarted (for example, upon a reboot).

#### Provisioning.SshHostKeyPairType:

Type: String  
Default: rsa

This can be set to an encryption algorithm type that is supported by the SSH daemon on the virtual machine. The typically supported values are "rsa", "dsa" and "ecdsa". "putty.exe" on Windows does not support "ecdsa". So, if you intend to use putty.exe on Windows to connect to a Linux deployment, use "rsa" or "dsa".

#### Provisioning.MonitorHostName:

Type: Boolean  
Default: y

If set, waagent monitors the Linux virtual machine for hostname changes (as returned by the "hostname" command) and automatically update the networking configuration in the image to reflect the change. In order to push the name change to the DNS servers, networking is restarted in the virtual machine. This results in brief loss of Internet connectivity.

## **Provisioning.DecodeCustomData**

Type: Boolean  
Default: n

If set, waagent decodes CustomData from Base64.

## **Provisioning.ExecuteCustomData**

Type: Boolean  
Default: n

If set, waagent executes CustomData after provisioning.

## **Provisioning.AllowResetSysUser**

Type: Boolean  
Default: n

This option allows the password for the sys user to be reset; default is disabled.

## **Provisioning.PasswordCryptId**

Type: String  
Default: 6

Algorithm used by crypt when generating password hash.

- 1 - MD5
- 2a - Blowfish
- 5 - SHA-256
- 6 - SHA-512

## **Provisioning.PasswordCryptSaltLength**

Type: String  
Default: 10

Length of random salt used when generating password hash.

## **ResourceDisk.Format:**

Type: Boolean  
Default: y

If set, the resource disk provided by the platform is formatted and mounted by waagent if the filesystem type requested by the user in "ResourceDisk.Filesystem" is anything other than "ntfs". A single partition of type Linux (83) is made available on the disk. This partition is not formatted if it can be successfully mounted.

## **ResourceDisk.Filesystem:**

Type: String  
Default: ext4

This specifies the filesystem type for the resource disk. Supported values vary by Linux distribution. If the string is X, then mkfs.X should be present on the Linux image. SLES 11 images should typically use 'ext3'. FreeBSD images should use 'ufs2' here.

#### **ResourceDisk.MountPoint:**

```
Type: String  
Default: /mnt/resource
```

This specifies the path at which the resource disk is mounted. The resource disk is a *temporary* disk, and might be emptied when the VM is deprovisioned.

#### **ResourceDisk.MountOptions**

```
Type: String  
Default: None
```

Specifies disk mount options to be passed to the mount -o command. This is a comma-separated list of values, ex. 'nodev,nosuid'. See mount(8) for details.

#### **ResourceDisk.EnableSwap:**

```
Type: Boolean  
Default: n
```

If set, a swap file (/swapfile) is created on the resource disk and added to the system swap space.

#### **ResourceDisk.SwapSizeMB:**

```
Type: Integer  
Default: 0
```

The size of the swap file in megabytes.

#### **Logs.Verbose:**

```
Type: Boolean  
Default: n
```

If set, log verbosity is boosted. Waagent logs to `/var/log/waagent.log` and utilizes the system logrotate functionality to rotate logs.

#### **OS.EnableRDMA**

```
Type: Boolean  
Default: n
```

If set, the agent attempts to install and then load an RDMA kernel driver that matches the version of the firmware on the underlying hardware.

#### **OS.RootDeviceScsiTimeout:**

Type: Integer  
Default: 300

This setting configures the SCSI timeout in seconds on the OS disk and data drives. If not set, the system defaults are used.

#### OS.OpensslPath:

Type: String  
Default: None

This setting can be used to specify an alternate path for the openssl binary to use for cryptographic operations.

#### HttpProxy.Host, HttpProxy.Port

Type: String  
Default: None

If set, the agent uses this proxy server to access the internet.

#### AutoUpdate.Enabled

Type: Boolean  
Default: y

Enable or disable auto-update for goal state processing; default is enabled.

## Linux Guest Agent Automatic Logs Collection

As of version 2.7+, The azure linux guest agent has a feature to automatically collect some logs and upload them. This feature currently requires systemd, and utilizes a new systemd slice called `azure-walinuxagent-logcollector.slice` to manage resources while performing the collection. The log collector's goal is facilitate offline analysis, and therefore produces a ZIP file of some diagnostics logs before uploading them to the VM's Host. The ZIP file can then be retrieved by Engineering Teams and Support professionals to investigate issues at the behest of the VM owner. More technical information on the files collected by the guest agent can be found in the [azurelinuxagent/common/logcollector\\_manifests.py](#) file in the [agent's GitHub repository](#).

This can be disabled by editing `/etc/waagent.conf` updating `Logs.Collect` to `n`

## Ubuntu Cloud Images

Ubuntu Cloud Images utilize [cloud-init](#) to perform many configuration tasks that would otherwise be managed by the Azure Linux Agent. The following differences apply:

- **Provisioning.Enabled** defaults to "n" on Ubuntu Cloud Images that use cloud-init to perform provisioning tasks.
- The following configuration parameters have no effect on Ubuntu Cloud Images that use cloud-init to manage the resource disk and swap space:
  - **ResourceDisk.Format**
  - **ResourceDisk.Filesystem**
  - **ResourceDisk.MountPoint**
  - **ResourceDisk.EnableSwap**

- **ResourceDisk.SwapSizeMB**
- For more information, see the following resources to configure the resource disk mount point and swap space on Ubuntu Cloud Images during provisioning:
  - [Ubuntu Wiki: Configure Swap Partitions](#)
  - [Injecting Custom Data into an Azure Virtual Machine](#)

# Virtual machine extensions and features for Windows

9/21/2022 • 12 minutes to read • [Edit Online](#)

Azure virtual machine (VM) extensions are small applications that provide post-deployment configuration and automation tasks on Azure VMs. For example, if a virtual machine requires software installation, antivirus protection, or the ability to run a script inside it, you can use a VM extension.

You can run Azure VM extensions by using the Azure CLI, PowerShell, Azure Resource Manager templates (ARM templates), and the Azure portal. You can bundle extensions with a new VM deployment or run them against any existing system.

This article provides an overview of Azure VM extensions, prerequisites for using them, and guidance on how to detect, manage, and remove them. This article provides generalized information because many VM extensions are available. Each has a potentially unique configuration and its own documentation.

## Use cases and samples

Each Azure VM extension has a specific use case. Examples include:

- Apply PowerShell desired state configurations (DSCs) to a VM by using the [DSC extension for Windows](#).
- Configure monitoring of a VM by using the [Log Analytics Agent VM extension](#).
- Configure an Azure VM by using [Chef](#).
- Configure monitoring of your Azure infrastructure by using the [Datadog extension](#).

In addition to process-specific extensions, a Custom Script extension is available for both Windows and Linux virtual machines. The [Custom Script extension for Windows](#) allows any PowerShell script to be run on a VM.

Custom scripts are useful for designing Azure deployments that require configuration beyond what native Azure tooling can provide.

## Prerequisites

### Azure VM Agent

To handle the extension on the VM, you need the [Azure VM Agent for Windows](#) (also called the Windows Guest Agent) installed. Some individual extensions have prerequisites, such as access to resources or dependencies.

The Azure VM Agent manages interactions between an Azure VM and the Azure fabric controller. The agent is responsible for many functional aspects of deploying and managing Azure VMs, including running VM extensions.

The Azure VM Agent is preinstalled on Azure Marketplace images. It can also be installed manually on supported operating systems.

The agent runs on multiple operating systems. However, the extensions framework has a [limit for the operating systems that extensions use](#). Some extensions are not supported across all operating systems and might emit error code 51 ("Unsupported OS"). Check the individual extension documentation for supportability.

### Network access

Extension packages are downloaded from the Azure Storage extension repository. Extension status uploads are posted to Azure Storage.

If you use a [supported version of the Azure VM Agent](#), you don't need to allow access to Azure Storage in the VM region. You can use the agent to redirect the communication to the Azure fabric controller for agent communications (HostGAPPlugin feature through the privileged channel on private IP [168.63.129.16](#)). If you're on an unsupported version of the agent, you need to allow outbound access to Azure Storage in that region from the VM.

#### IMPORTANT

If you've blocked access to 168.63.129.16 by using the guest firewall or by using a proxy, extensions fail even if you're using a supported version of the agent or you've configured outbound access. Ports 80, 443, and 32526 are required.

Agents can only be used to download extension packages and reporting status. For example, if an extension installation needs to download a script from GitHub (Custom Script extension) or needs access to Azure Storage (Azure Backup), then you need to open additional firewall or network security group (NSG) ports. Different extensions have different requirements, because they're applications in their own right. For extensions that require access to Azure Storage or Azure Active Directory, you can allow access by using Azure NSG [service tags](#).

The Azure VM Agent does not have proxy server support for you to redirect agent traffic requests through. That means the Azure VM Agent will rely on your custom proxy (if you have one) to access resources on the internet or on the host through IP 168.63.129.16.

## Discover VM extensions

Many VM extensions are available for use with Azure VMs. To see a complete list, use [Get-AzVMExtensionImage](#). The following example lists all available extensions in the *WestUS* location:

```
Get-AzVmImagePublisher -Location "WestUS" |  
Get-AzVMExtensionImageType |  
Get-AzVMExtensionImage | Select Type, Version
```

## Run VM extensions

Azure VM extensions run on existing VMs. That's useful when you need to make configuration changes or recover connectivity on an already deployed VM. VM extensions can also be bundled with ARM template deployments. By using extensions with ARM templates, you can deploy and configure Azure VMs without post-deployment intervention.

You can use the following methods to run an extension against an existing VM.

#### PowerShell

Several PowerShell commands exist for running individual extensions. To see a list, use [Get-Command](#) and filter on *Extension*:

```
Get-Command Set-Az*Extension* -Module Az.Compute
```

This command provides output similar to the following:

CommandType	Name	Version	Source
Cmdlet	Set-AzVMAccessExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMAccessExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMAEMExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMBackupExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMBinfoExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMChefExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMCustomScriptExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMDiagnosticsExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMDiskEncryptionExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMDscExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMEExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVMSqlServerExtension	4.5.0	Az.Compute
Cmdlet	Set-AzVmssDiskEncryptionExtension	4.5.0	Az.Compute

The following example uses the [Custom Script extension](#) to download a script from a GitHub repository onto the target virtual machine and then run the script:

```
Set-AzVMCustomScriptExtension -ResourceGroupName "myResourceGroup" ` 
    -VMName "myVM" -Name "myCustomScript" ` 
    -FileUri "https://raw.githubusercontent.com/neilpeterson/nepeters-azure-templates/master/windows-custom-` 
    -script-simple/support-scripts/Create-File.ps1" ` 
    -Run "Create-File.ps1" -Location "West US"
```

The following example uses the [VMAccess extension](#) to reset the administrative password of a Windows VM to a temporary password. After you run this code, you should reset the password at first login.

```
$cred=Get-Credential

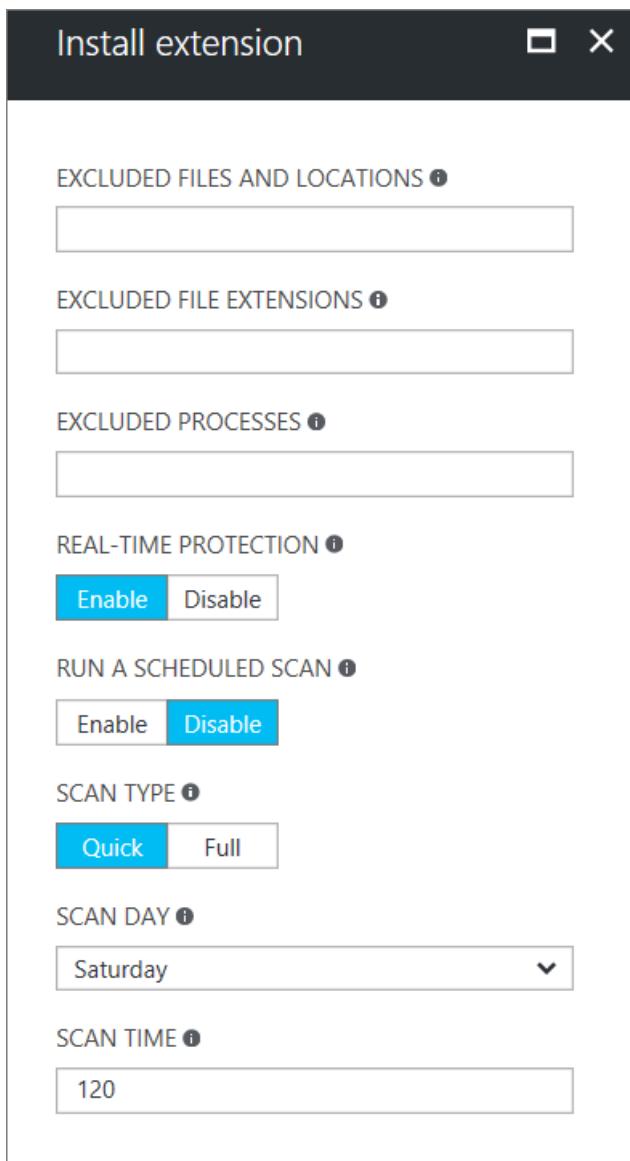
Set-AzVMAccessExtension -ResourceGroupName "myResourceGroup" -VMName "myVM" -Name "myVMAccess" ` 
    -Location WestUS -UserName $cred.GetNetworkCredential().Username ` 
    -Password $cred.GetNetworkCredential().Password -typeHandlerVersion "2.0"
```

You can use the [Set-AzVMEExtension](#) command to start any VM extension.

## Azure portal

You can apply VM extensions to an existing VM through the Azure portal. Select the VM in the portal, select **Extensions**, and then select **Add**. Choose the extension that you want from the list of available extensions, and follow the instructions in the wizard.

The following example shows the installation of the Microsoft Antimalware extension from the Azure portal:



### Azure Resource Manager templates

You can add VM extensions to an ARM template and run them with the deployment of the template. When you deploy an extension with a template, you can create fully configured Azure deployments.

For example, the following JSON is taken from a [full ARM template](#) that deploys a set of load-balanced VMs and an Azure SQL database, and then installs a .NET Core application on each VM. The VM extension takes care of the software installation.

```
{
  "apiVersion": "2015-06-15",
  "type": "extensions",
  "name": "config-app",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'),copyindex())]",
    "[variables('musicstoresqlName')]"
  ],
  "tags": {
    "displayName": "config-app"
  },
  "properties": {
    "publisher": "Microsoft.Compute",
    "type": "CustomScriptExtension",
    "typeHandlerVersion": "1.9",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "fileUris": [
        "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1"
      ]
    },
    "protectedSettings": {
      "commandToExecute": "[concat('powershell -ExecutionPolicy Unrestricted -File configure-music-app.ps1 -user ',parameters('adminUsername'),' -password ',parameters('adminPassword'),' -sqlserver ',variables('musicstoresqlName'),'.database.windows.net')]"
    }
  }
}
```

For more information on creating ARM templates, see [Virtual machines in an Azure Resource Manager template](#).

## Help secure VM extension data

When you run a VM extension, it might be necessary to include sensitive information such as credentials, storage account names, and access keys. Many VM extensions include a protected configuration that encrypts data and only decrypts it inside the target VM. Each extension has a specific protected configuration schema, and each is detailed in extension-specific documentation.

The following example shows an instance of the Custom Script extension for Windows. The command to run includes a set of credentials. In this example, the command to run is not encrypted.

```
{
    "apiVersion": "2015-06-15",
    "type": "extensions",
    "name": "config-app",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'),copyIndex())]",
        "[variables('musicstoresqlName')]"
    ],
    "tags": {
        "displayName": "config-app"
    },
    "properties": {
        "publisher": "Microsoft.Compute",
        "type": "CustomScriptExtension",
        "typeHandlerVersion": "1.9",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "fileUris": [
                "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1"
            ],
            "commandToExecute": "[concat('powershell -ExecutionPolicy Unrestricted -File configure-music-app.ps1 -user ',parameters('adminUsername'),' -password ',parameters('adminPassword'),' -sqlserver ',variables('musicstoresqlName'),'.database.windows.net')]"
        }
    }
}
```

Moving the `commandToExecute` property to the `protected` configuration helps secure the execution string, as shown in the following example:

```
{
    "apiVersion": "2015-06-15",
    "type": "extensions",
    "name": "config-app",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'),copyIndex())]",
        "[variables('musicstoresqlName')]"
    ],
    "tags": {
        "displayName": "config-app"
    },
    "properties": {
        "publisher": "Microsoft.Compute",
        "type": "CustomScriptExtension",
        "typeHandlerVersion": "1.9",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "fileUris": [
                "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1"
            ]
        },
        "protectedSettings": {
            "commandToExecute": "[concat('powershell -ExecutionPolicy Unrestricted -File configure-music-app.ps1 -user ',parameters('adminUsername'),' -password ',parameters('adminPassword'),' -sqlserver ',variables('musicstoresqlName'),'.database.windows.net')]"
        }
    }
}
```

On an Azure infrastructure as a service (IaaS) VM that uses extensions, in the certificates console, you might see

certificates that have the subject **Windows Azure CRP Certificate Generator**. On a classic RedDog Front End (RDDE) VM, these certificates have the subject name **Windows Azure Service Management for Extensions**.

These certificates secure the communication between the VM and its host during the transfer of protected settings (password and other credentials) that extensions use. The certificates are built by the Azure fabric controller and passed to the Azure VM Agent. If you stop and start the VM every day, the fabric controller might create a new certificate. The certificate is stored in the computer's personal certificate store. These certificates can be deleted. The Azure VM Agent re-creates certificates if needed.

## How agents and extensions are updated

Agents and extensions share the same automatic update mechanism.

When an update is available and automatic updates are enabled, the update is installed on the VM only after there's a change to an extension or after other VM model changes, such as:

- Data disks
- Extensions
- Extension Tags
- Boot diagnostics container
- Guest OS secrets
- VM size
- Network profile

Publishers make updates available to regions at various times, so it's possible that you can have VMs in different regions on different versions.

### NOTE

Some updates might require additional firewall rules. See [Network access](#).

## Listing extensions deployed to a VM

```
$vm = Get-AzVM -ResourceGroupName "myResourceGroup" -VMName "myVM"  
$vm.Extensions | select Publisher, VirtualMachineExtensionType, TypeHandlerVersion
```

Publisher	VirtualMachineExtensionType	TypeHandlerVersion
-----	-----	-----
Microsoft.Compute	CustomScriptExtension	1.9

## Agent updates

The Azure VM Agent contains only *extension-handling code*. The *Windows provisioning code* is separate. You can uninstall the Azure VM Agent. You can't disable the automatic update of the Azure VM Agent.

The extension-handling code is responsible for:

- Communicating with the Azure fabric.
- Handling the VM extension operations, such as installations, reporting status, updating the individual extensions, and removing extensions. Updates contain security fixes, bug fixes, and enhancements to the extension-handling code.

To check what version you're running, see [Detect the VM Agent](#).

## Extension updates

When an extension update is available and automatic updates are enabled, after a [change to the VM model](#) occurs, the Azure VM Agent downloads and upgrades the extension.

Automatic extension updates are either *minor* or *hotfix*. You can opt in or opt out of minor updates when you provision the extension. The following example shows how to automatically upgrade minor versions in an ARM template by using `"autoUpgradeMinorVersion": true,`:

```
"properties": {  
    "publisher": "Microsoft.Compute",  
    "type": "CustomScriptExtension",  
    "typeHandlerVersion": "1.9",  
    "autoUpgradeMinorVersion": true,  
    "settings": {  
        "fileUris": [  
            "https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1"  
        ]  
    },  
},
```

To get the latest minor-release bug fixes, we highly recommend that you always select automatic update in your extension deployments. You can't opt out of hotfix updates that carry security or key bug fixes.

If you disable automatic updates or you need to upgrade a major version, use [Set-AzVMEExtension](#) and specify the target version.

## How to identify extension updates

### Identify if the extension is set with `autoUpgradeMinorVersion` on a VM

You can see from the VM model if the extension was provisioned with `autoUpgradeMinorVersion`. To check, use [Get-AzVm](#) and provide the resource group and VM name as follows:

```
$vm = Get-AzVm -ResourceGroupName "myResourceGroup" -VMName "myVM"  
$vm.Extensions
```

The following example output shows that `autoUpgradeMinorVersion` is set to `true`:

```
ForceUpdateTag :  
Publisher : Microsoft.Compute  
VirtualMachineExtensionType : CustomScriptExtension  
TypeHandlerVersion : 1.9  
AutoUpgradeMinorVersion : True
```

### Identify when an `autoUpgradeMinorVersion` event occurred

To see when an update to the extension occurred, review the agent logs on the VM at `C:\WindowsAzure\Logs\WaAppAgent.log`.

In the following example, the VM had `Microsoft.Compute.CustomScriptExtension` version `1.8` installed. A hotfix was available to version `1.9`.

```
[INFO] Getting plugin locations for plugin 'Microsoft.Compute.CustomScriptExtension'. Current Version:  
'1.8', Requested Version: '1.9'  
[INFO] Auto-Upgrade mode. Highest public version for plugin 'Microsoft.Compute.CustomScriptExtension' with  
requested version: '1.9', is: '1.9'
```

## Agent permissions

To perform its tasks, the agent needs to run as *Local System*.

## Troubleshoot VM extensions

Each VM extension might have specific troubleshooting steps. For example, when you use the Custom Script extension, you can find script execution details locally on the VM where the extension was run.

The following troubleshooting actions apply to all VM extensions:

- To check the Azure VM Agent Log, look at the activity when your extension was being provisioned in *C:\WindowsAzure\Logs\WaAppAgent.log*.
- Check the extension logs for more details in *C:\WindowsAzure\Logs\Plugins<extensionName>*.
- Check troubleshooting sections in extension-specific documentation for error codes, known issues, and other extension-specific information.
- Look at the system logs. Check for other operations that might have interfered with the extension, such as a long-running installation of another application that required exclusive access to the package manager.
- In a VM, if there is an existing extension with a failed provisioning state, any other new extension fails to install.

### Common reasons for extension failures

- Extensions have 20 minutes to run. (Exceptions are Custom Script, Chef, and DSC, which have 90 minutes.) If your deployment exceeds this time, it's marked as a timeout. The cause of this can be low-resource VMs, or other VM configurations or startup tasks are consuming large amounts of resources while the extension is trying to provision.
- Minimum prerequisites aren't met. Some extensions have dependencies on VM SKUs, such as HPC images. Extensions might have certain networking access requirements, such as communicating with Azure Storage or public services. Other examples might be access to package repositories, running out of disk space, or security restrictions.
- Package manager access is exclusive. In some cases, a long-running VM configuration and extension installation might conflict because they both need exclusive access to the package manager.

### View extension status

After a VM extension has been run against a VM, use [Get-AzVM](#) to return extension status. `Substatuses[0]` shows that the extension provisioning succeeded, meaning that it successfully deployed to the VM. But `Substatuses[1]` shows that the execution of the extension inside the VM failed.

```
Get-AzVM -ResourceGroupName "myResourceGroup" -VMName "myVM" -Status
```

The output is similar to the following example:

```

Extensions[0]      :
  Name          : CustomScriptExtension
  Type          : Microsoft.Compute.CustomScriptExtension
  TypeHandlerVersion : 1.9
  Substatuses[0]   :
    Code        : ComponentStatus/StdOut/succeeded
    Level       : Info
    DisplayStatus: Provisioning succeeded
    Message     : Windows PowerShell \nCopyright (C) Microsoft Corporation. All rights reserved.\n
  Substatuses[1]   :
    Code        : ComponentStatus/StdErr/succeeded
    Level       : Info
    DisplayStatus: Provisioning succeeded
    Message     : The argument 'cseTest%20Scriptparam1.ps1' to the -File parameter does not exist.
Provide the path to an existing '.ps1' file as an argument to the
-File parameter.
  Statuses[0]     :
    Code          : ProvisioningState/failed/-196608
    Level         : Error
    DisplayStatus : Provisioning failed
    Message       : Finished executing command

```

You can also find extension execution status in the Azure portal. Select the VM, select **Extensions**, and then select the desired extension.

### Rerun a VM extension

There might be cases in which a VM extension needs to be rerun. You can rerun an extension by removing it, and then rerunning the extension with an execution method of your choice. To remove an extension, use [Remove-AzVMExtension](#) as follows:

```
Remove-AzVMExtension -ResourceGroupName "myResourceGroup" -VMName "myVM" -Name "myExtensionName"
```

You can also remove an extension in the Azure portal:

1. Select a VM.
2. Select **Extensions**.
3. Select the extension.
4. Select **Uninstall**.

## Common VM extension reference

EXTENSION NAME	DESCRIPTION
<a href="#">Custom Script extension for Windows</a>	Run scripts against an Azure virtual machine.
<a href="#">DSC extension for Windows</a>	Apply PowerShell desired state configurations to a virtual machine.
<a href="#">Azure Diagnostics extension</a>	Manage Azure Diagnostics.
<a href="#">VMAccess extension</a>	Manage users and credentials.

## Next steps

For more information about VM extensions, see [Azure virtual machine extensions and features](#).

# Azure Virtual Machine Agent overview

9/21/2022 • 5 minutes to read • [Edit Online](#)

The Microsoft Azure Virtual Machine Agent (VM Agent) is a secure, lightweight process that manages virtual machine (VM) interaction with the Azure Fabric Controller. The VM Agent has a primary role in enabling and executing Azure virtual machine extensions. VM Extensions enable post-deployment configuration of VM, such as installing and configuring software. VM extensions also enable recovery features such as resetting the administrative password of a VM. Without the Azure VM Agent, VM extensions cannot be run.

This article details installation and detection of the Azure Virtual Machine Agent.

## Install the VM Agent

### Azure Marketplace image

The Azure VM Agent is installed by default on any Windows VM deployed from an Azure Marketplace image. When you deploy an Azure Marketplace image from the portal, PowerShell, Command Line Interface, or an Azure Resource Manager template, the Azure VM Agent is also installed.

The Windows Guest Agent Package is broken into two parts:

- Provisioning Agent (PA)
- Windows Guest Agent (WinGA)

To boot a VM you must have the PA installed on the VM, however the WinGA does not need to be installed. At VM deploy time, you can select not to install the WinGA. The following example shows how to select the *provisionVmAgent* option with an Azure Resource Manager template:

```
{
  "resources": [
    {
      "name": "[parameters('virtualMachineName')]",
      "type": "Microsoft.Compute/virtualMachines",
      "apiVersion": "2016-04-30-preview",
      "location": "[parameters('location')]",
      "dependsOn": "[[concat('Microsoft.Network/networkInterfaces/', parameters('networkInterfaceName'))]]",
      "properties": {
        "osProfile": {
          "computerName": "[parameters('virtualMachineName')]",
          "adminUsername": "[parameters('adminUsername')]",
          "adminPassword": "[parameters('adminPassword')]",
          "windowsConfiguration": {
            "provisionVmAgent": "false"
          }
        }
      }
    }
  ]
}
```

If you do not have the Agents installed, you cannot use some Azure services, such as Azure Backup or Azure Security. These services require an extension to be installed. If you have deployed a VM without the WinGA, you can install the latest version of the agent later.

### Manual installation

The Windows VM agent can be manually installed with a Windows installer package. Manual installation may be necessary when you create a custom VM image that is deployed to Azure. To manually install the Windows VM

Agent, [download the VM Agent installer](#) and select the latest release. You can also search a specific version in the [GitHub Windows IaaS VM Agent releases](#). The VM Agent is supported on Windows Server 2008 (64 bit) and later.

#### NOTE

It is important to update the AllowExtensionOperations option after manually installing the VMAgent on a VM that was deployed from image without ProvisionVMAgent enable.

```
$vm.OSProfile.AllowExtensionOperations = $true  
$vm | Update-AzVM
```

## Prerequisites

- The Windows VM Agent needs at least Windows Server 2008 SP2 (64-bit) to run, with the .NET Framework 4.0. See [Minimum version support for virtual machine agents in Azure](#).
- Ensure your VM has access to IP address 168.63.129.16. For more information, see [What is IP address 168.63.129.16](#).
- Ensure that DHCP is enabled inside the guest VM. This is required to get the host or fabric address from DHCP for the IaaS VM Agent and extensions to work. If you need a static private IP, you should configure it through the Azure portal or PowerShell, and make sure the DHCP option inside the VM is enabled. [Learn more](#) about setting up a static IP address with PowerShell.
- Running the VM Agent in a "Nested Virtualization" VM might lead to unpredictable behavior, hence it's not supported in that Dev/Test scenario.

## Detect the VM Agent

### PowerShell

The Azure Resource Manager PowerShell module can be used to retrieve information about Azure VMs. To see information about a VM, such as the provisioning state for the Azure VM Agent, use [Get-AzVM](#):

```
Get-AzVM
```

The following condensed example output shows the *ProvisionVMAgent* property nested inside `OSProfile`. This property can be used to determine if the VM agent has been deployed to the VM:

```
OSProfile :  
  ComputerName : myVM  
  AdminUsername : myUserName  
  WindowsConfiguration :  
    ProvisionVMAgent : True  
    EnableAutomaticUpdates : True
```

The following script can be used to return a concise list of VM names (running Windows OS) and the state of the VM Agent:

```

$vms = Get-AzVM

foreach ($vm in $vms) {
    $agent = $vm | Select -ExpandProperty OSProfile | Select -ExpandProperty WindowsConfiguration | Select
    ProvisionVMAgent
    Write-Host $vm.Name $agent.ProvisionVMAgent
}

```

The following script can be used to return a concise list of VM names (running Linux OS) and the state of the VM Agent:

```

$vms = Get-AzVM

foreach ($vm in $vms) {
    $agent = $vm | Select -ExpandProperty OSProfile | Select -ExpandProperty LinuxConfiguration | Select
    ProvisionVMAgent
    Write-Host $vm.Name $agent.ProvisionVMAgent
}

```

## Manual Detection

When logged in to a Windows VM, Task Manager can be used to examine running processes. To check for the Azure VM Agent, open Task Manager, click the *Details* tab, and look for a process name `WindowsAzureGuestAgent.exe`. The presence of this process indicates that the VM agent is installed.

## Upgrade the VM Agent

The Azure VM Agent for Windows is automatically upgraded on images deployed from the Azure Marketplace. The new versions are stored in Azure Storage, so please ensure you don't have firewalls blocking access. As new VMs are deployed to Azure, they receive the latest VM agent at VM provision time. If you have installed the agent manually or are deploying custom VM images you will need to manually update to include the new VM agent at image creation time.

## Windows Guest Agent Automatic Logs Collection

Windows Guest Agent has a feature to automatically collect some logs. This feature is controlled by the `CollectGuestLogs.exe` process. It exists for both PaaS Cloud Services and IaaS Virtual Machines and its goal is to quickly & automatically collect some diagnostics logs from a VM - so they can be used for offline analysis. The collected logs are Event Logs, OS Logs, Azure Logs and some registry keys. It produces a ZIP file that is transferred to the VM's Host. This ZIP file can then be looked at by Engineering Teams and Support professionals to investigate issues on request of the customer owning the VM.

## Guest Agent and OSProfile certificates

The Azure VM Agent is responsible for installing the certificates referenced in the `OSProfile` of a VM or Virtual Machine Scale Set. If you manually remove these certificates from the certificates MMC console inside the guest VM, it is expected that the guest agent will add them back. To permanently remove a certificate, you will have to remove it from the `OSProfile`, and then remove it from within the guest operating system.

For a Virtual Machine, use the [Remove-AzVMSecret](#) to remove certificates from the `OSProfile`.

For more information on Virtual Machine Scale Set certificates, see [Virtual Machine Scale Sets - How do I remove deprecated certificates?](#)

## Next steps

For more information about VM extensions, see [Azure virtual machine extensions and features overview](#).

# Azure Backup for SQL Server running in Azure VM

9/21/2022 • 2 minutes to read • [Edit Online](#)

Azure Backup, amongst other offerings, provides support for backing up workloads such as SQL Server running in Azure VMs. Since the SQL application is running within an Azure VM, the backup service needs permission to access the application and fetch the necessary details. To do that, Azure Backup installs the **AzureBackupWindowsWorkload** extension on the VM, in which the SQL Server is running, during the registration process triggered by the user.

## Prerequisites

For the list of supported scenarios, refer to the [supportability matrix](#) supported by Azure Backup.

## Network connectivity

Azure Backup supports NSG Tags, deploying a proxy server or listed IP ranges; for details on each of the methods, refer this [article](#).

## Extension schema

The extension schema and property values are the configuration values (runtime settings) that service is passing to CRP API. These config values are used during registration and upgrade. **AzureBackupWindowsWorkload** extension also uses this schema. The schema is pre-set; a new parameter can be added in the objectStr field

```
"runtimeSettings": [{"handlerSettings": {"protectedSettingsCertThumbprint": "", "protectedSettings": {"objectStr": "", "logsBlobUri": "", "statusBlobUri": ""}}, "publicSettings": {"locale": "en-us", "taskId": "1c0ae461-9d3b-418c-a505-bb31dfe2095d", "objectStr": "", "commandStartTimeUTCTicks": "636295005824665976", "vmType": "vmType"}}], "objectStr": ""}]}
```

The following JSON shows the schema for the WorkloadBackup extension.

```
{
  "type": "extensions",
  "name": "WorkloadBackup",
  "location": "<myLocation>",
  "properties": {
    "publisher": "Microsoft.RecoveryServices",
    "type": "AzureBackupWindowsWorkload",
    "typeHandlerVersion": "1.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "locale": "<location>",
      "taskId": "<TaskId used by Azure Backup service to communicate with extension>",

      "objectStr": "<The configuration passed by Azure Backup service to extension>",

      "commandStartTimeUTCTicks": "<Scheduled start time of registration or upgrade task>",
      "vmType": "<Type of VM where registration got triggered Eg. Compute or ClassicCompute>"
    },
    "protectedSettings": {
      "objectStr": "<The sensitive configuration passed by Azure Backup service to extension>",
      "logsBlobUri": "<blob uri where logs of command execution by extension are written to>",
      "statusBlobUri": "<blob uri where status of the command executed by extension is written>"
    }
  }
}
```

## Property values

NAME	VALUE/EXAMPLE	DATA TYPE
locale	en-us	string
taskId	"1c0ae461-9d3b-418c-a505-bb31dfe2095d"	string
objectStr (publicSettings)	"eyJjb250YWluZXJQcm9wZXJ0aWVzIjp7IkNvbRhaW5lcklEljoiMzVjMjQxYTItOGRjNy00ZGE5LWI4NTMtMjdjYTJhNDZlM2ZkliwiSWRNZ210Q29udGFpbmVySWQiOjM0NTY3ODg5LCJSZXNvdXJjZUIkljoiMDU5NWlwOGEtYzI4Zi00ZmFILWE5ODItOTkwOWMyMGVjNjVhliwiU3Vi c2NyaXB0aW9uSWQiOijkNGEzOTliNy1iYjAyLTQ2MWMtODdmYS1jNTM5ODI3ZTgzNTQiLCJVbmlxdWVDb250YWluZXJOYW1lIjoiODM4MDZjODUtNTQ4OS00NmNhLWEyZTctNWMzNzNhYjg3OTcyIn0sInN0YW1wTGlzdCI6W3siU2VydmljZU5hbWUiOjUsInIcnZpY2VTdGFtcFVybCI6Imh0dHA6XC9cL015V0xGYWJTdmMuY29tIn1dfQ=="	string
commandStartTimeUTCTicks	"636967192566036845"	string
vmType	"microsoft.compute/virtualmachines"	string

Name	Value/Example	Data Type
objectStr (protectedSettings)	"eyJjb250YWluZXJQcm9wZXJ0aWVzIjp7IkNvbRhaW5lcklEljoiMzVjMjQxYTItOGRjNy00ZGE5LWI4NTMtMjdjYTJhNDZIM2ZkliwiSWRNZ210Q29udGFpbmVySWQiOjM0NTY3ODg5LCJSZXNvdXJjZUIkljoiMDU5NWlwOGEtYzI4Zl00ZmFILWE5ODItOTkwOWMyMGVjNjVhliwiU3ViC2NyaXB0aW9uSWQiOjlkNGEzOTliNy1iYjAyLTQ2MWMTODdmYS1jNTM5ODI3ZTgzNTQiLCJVbmlxdWVDb250YWluZXJOYW1lljoiODM4MDZjODUtNTQ4OS00NmNhLWEyZTctNWMzNzNhYjg3OTcyIn0slnN0YW1wtGlzdCI6W3siU2VydmljZU5hbWUiOjUsIINlcnPjY2VTdGFtcFVybCI6Imh0dHA6XC9cL015V0xGYWJTdmMuY29tIn1dfQ=="	string
logsBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DBwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DBwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string
statusBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5lduB%2FlktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5lduB%2FlktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string

## Template deployment

We recommended adding AzureBackupWindowsWorkload extension to a virtual machine by enabling SQL Server backup on the virtual machine. This can be achieved through the [Resource Manager template](#) designed for automating backup on a SQL Server VM.

## PowerShell deployment

You need to register the Azure VM that contains the SQL application with a Recovery services vault. During registration, AzureBackupWindowsWorkload extension gets installed on the VM. Use [Register-AzRecoveryServicesBackupContainerPS](#) cmdlet to register the VM.

```
$myVM = Get-AzVM -ResourceGroupName <VMRG Name> -Name <VMName>
Register-AzRecoveryServicesBackupContainer -ResourceId $myVM.ID -BackupManagementType AzureWorkload -WorkloadType MSSQL -VaultId $targetVault.ID -Force
```

The command will return a **backup container** of this resource and the status will be **registered**.

## Next steps

- [Learn More](#) about Azure SQL Server VM backup troubleshooting guidelines
- [Common questions](#) about backing up SQL Server databases that run on Azure virtual machines (VMs) and that use the Azure Backup service.

# Azure Disk Encryption for Linux (Microsoft.Azure.Security.AzureDiskEncryptionForLinux)

9/21/2022 • 3 minutes to read • [Edit Online](#)

## Overview

Azure Disk Encryption leverages the dm-crypt subsystem in Linux to provide full disk encryption on [select Azure Linux distributions](#). This solution is integrated with Azure Key Vault to manage disk encryption keys and secrets.

## Prerequisites

For a full list of prerequisites, see [Azure Disk Encryption for Linux VMs](#), specifically the following sections:

- [Supported VMs and operating systems](#)
- [Additional VM requirements](#)
- [Networking requirements](#)
- [Encryption key storage requirements](#)

## Extension Schema

There are two versions of extension schema for Azure Disk Encryption (ADE):

- v1.1 - A newer recommended schema that does not use Azure Active Directory (Azure AD) properties.
- v0.1 - An older schema that requires Azure Active Directory (Azure AD) properties.

To select a target schema, the `typeHandlerVersion` property must be set equal to version of schema you want to use.

### Schema v1.1: No Azure AD (recommended)

The v1.1 schema is recommended and does not require Azure Active Directory (Azure AD) properties.

#### NOTE

The `DiskFormatQuery` parameter is deprecated. Its functionality has been replaced by the `EncryptFormatAll` option instead, which is the recommended way to format data disks at time of encryption.

```
{
  "type": "extensions",
  "name": "[name]",
  "apiVersion": "2019-07-01",
  "location": "[location]",
  "properties": {
    "publisher": "Microsoft.Azure.Security",
    "type": "AzureDiskEncryptionForLinux",
    "typeHandlerVersion": "1.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "DiskFormatQuery": "[diskFormatQuery]",
      "EncryptionOperation": "[encryptionOperation]",
      "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",
      "KeyVaultURL": "[keyVaultURL]",
      "KeyVaultResourceId": "[KeyVaultResourceId]",
      "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",
      "KekVaultResourceId": "[KekVaultResourceId]",
      "SequenceVersion": "sequenceVersion",
      "VolumeType": "[volumeType]"
    }
  }
}
```

## Schema v0.1: with Azure AD

The 0.1 schema requires `AADClientID` and either `AADClientSecret` or `AADClientCertificate`.

Using `AADClientSecret`:

```
{
  "type": "extensions",
  "name": "[name]",
  "apiVersion": "2019-07-01",
  "location": "[location]",
  "properties": {
    "protectedSettings": {
      "AADClientSecret": "[aadClientSecret]",
      "Passphrase": "[passphrase]"
    },
    "publisher": "Microsoft.Azure.Security",
    "type": "AzureDiskEncryptionForLinux",
    "typeHandlerVersion": "0.1",
    "settings": {
      "AADClientID": "[aadClientID]",
      "DiskFormatQuery": "[diskFormatQuery]",
      "EncryptionOperation": "[encryptionOperation]",
      "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",
      "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",
      "KeyVaultURL": "[keyVaultURL]",
      "SequenceVersion": "sequenceVersion",
      "VolumeType": "[volumeType]"
    }
  }
}
```

Using `AADClientCertificate`:

```
{
  "type": "extensions",
  "name": "[name]",
  "apiVersion": "2019-07-01",
  "location": "[location]",
  "properties": {
    "protectedSettings": {
      "AADClientCertificate": "[aadClientCertificate]",
      "Passphrase": "[passphrase]"
    },
    "publisher": "Microsoft.Azure.Security",
    "type": "AzureDiskEncryptionForLinux",
    "typeHandlerVersion": "0.1",
    "settings": {
      "AADClientID": "[aadClientID]",
      "DiskFormatQuery": "[diskFormatQuery]",
      "EncryptionOperation": "[encryptionOperation]",
      "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",
      "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",
      "KeyVaultURL": "[keyVaultURL]",
      "SequenceVersion": "sequenceVersion",
      "VolumeType": "[volumeType]"
    }
  }
}
```

## Property values

Note: All property values are case sensitive.

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2019-07-01	date
publisher	Microsoft.Azure.Security	string
type	AzureDiskEncryptionForLinux	string
typeHandlerVersion	1.1, 0.1	int
(0.1 schema) AADClientID	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx	guid
(0.1 schema) AADClientSecret	password	string
(0.1 schema) AADClientCertificate	thumbprint	string
(optional) (0.1 schema) Passphrase	password	string
DiskFormatQuery	{"dev_path": "", "name": "", "file_system": ""}	JSON dictionary
EncryptionOperation	EnableEncryption, EnableEncryptionFormatAll	string
(optional - default RSA-OAEP ) KeyEncryptionAlgorithm	'RSA-OAEP', 'RSA-OAEP-256', 'RSA1_5'	string
KeyVaultURL	url	string

NAME	VALUE / EXAMPLE	DATA TYPE
KeyVaultResourceId	url	string
(optional) KeyEncryptionKeyURL	url	string
(optional) KekVaultResourceId	url	string
(optional) SequenceVersion	uniqueidentifier	string
VolumeType	OS, Data, All	string

## Template deployment

For an example of template deployment based on schema v1.1, see the Azure Quickstart Template [encrypt-running-linux-vm-without-aad](#).

For an example of template deployment based on schema v0.1, see the Azure Quickstart Template [encrypt-running-linux-vm](#).

### WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM.
- When encrypting Linux OS volumes, the VM should be considered unavailable. We strongly recommend to avoid SSH logins while the encryption is in progress to avoid issues blocking any open files that will need to be accessed during the encryption process. To check progress, use the `Get-AzVMDiskEncryptionStatus` PowerShell cmdlet or the `vm encryption show` CLI command. This process can be expected to take a few hours for a 30GB OS volume, plus additional time for encrypting data volumes. Data volume encryption time will be proportional to the size and quantity of the data volumes; the `encrypt format all` option is faster than in-place encryption, but will result in the loss of all data on the disks.
- Disabling encryption on Linux VMs is only supported for data volumes. It is not supported on data or OS volumes if the OS volume has been encrypted.

### NOTE

Also if `VolumeType` parameter is set to All, data disks will be encrypted only if they are properly mounted.

## Troubleshoot and support

### Troubleshoot

For troubleshooting, refer to the [Azure Disk Encryption troubleshooting guide](#).

### Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#).

Alternatively, you can file an Azure support incident. Go to [Azure support](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure Support FAQ](#).

## Next steps

- For more information about VM extensions, see [Virtual machine extensions and features for Linux](#).
- For more information about Azure Disk Encryption for Linux, see [Linux virtual machines](#).

# Azure Disk Encryption for Windows (Microsoft.Azure.Security.AzureDiskEncryption)

9/21/2022 • 2 minutes to read • [Edit Online](#)

## Overview

Azure Disk Encryption uses BitLocker to provide full disk encryption on Azure virtual machines running Windows. This solution is integrated with Azure Key Vault to manage disk encryption keys and secrets in your key vault subscription.

## Prerequisites

For a full list of prerequisites, see [Azure Disk Encryption for Windows VMs](#), specifically the following sections:

- [Supported VMs and operating systems](#)
- [Networking requirements](#)
- [Group Policy requirements](#)

## Extension Schema

There are two versions of extension schema for Azure Disk Encryption (ADE):

- v2.2 - A newer recommended schema that does not use Azure Active Directory (Azure AD) properties.
- v1.1 - An older schema that requires Azure Active Directory (Azure AD) properties.

To select a target schema, the `typeHandlerVersion` property must be set equal to version of schema you want to use.

### Schema v2.2: No Azure AD (recommended)

The v2.2 schema is recommended for all new VMs and does not require Azure Active Directory properties.

```
{  
    "type": "extensions",  
    "name": "[name]",  
    "apiVersion": "2019-07-01",  
    "location": "[location]",  
    "properties": {  
        "publisher": "Microsoft.Azure.Security",  
        "type": "AzureDiskEncryption",  
        "typeHandlerVersion": "2.2",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "EncryptionOperation": "[encryptionOperation]",  
            "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",  
            "KeyVaultURL": "[keyVaultURL]",  
            "KeyVaultResourceId": "[keyVaultResourceID]",  
            "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",  
            "KekVaultResourceId": "[kekVaultResourceID]",  
            "SequenceVersion": "sequenceVersion]",  
            "VolumeType": "[volumeType]"  
        }  
    }  
}
```

## Schema v1.1: with Azure AD

The 1.1 schema requires `aadClientID` and either `aadClientSecret` or `AADClientCertificate` and is not recommended for new VMs.

Using `aadClientSecret`:

```
{
  "type": "extensions",
  "name": "[name]",
  "apiVersion": "2019-07-01",
  "location": "[location]",
  "properties": {
    "protectedSettings": {
      "AADClientSecret": "[aadClientSecret]"
    },
    "publisher": "Microsoft.Azure.Security",
    "type": "AzureDiskEncryption",
    "typeHandlerVersion": "1.1",
    "settings": {
      "AADClientID": "[aadClientID]",
      "EncryptionOperation": "[encryptionOperation]",
      "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",
      "KeyVaultURL": "[keyVaultURL]",
      "KeyVaultResourceId": "[keyVaultResourceID]",
      "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",
      "KekVaultResourceId": "[kekVaultResourceID]",
      "SequenceVersion": "sequenceVersion",
      "VolumeType": "[volumeType]"
    }
  }
}
```

Using `AADClientCertificate`:

```
{
  "type": "extensions",
  "name": "[name]",
  "apiVersion": "2019-07-01",
  "location": "[location]",
  "properties": {
    "protectedSettings": {
      "AADClientCertificate": "[aadClientCertificate]"
    },
    "publisher": "Microsoft.Azure.Security",
    "type": "AzureDiskEncryption",
    "typeHandlerVersion": "1.1",
    "settings": {
      "AADClientID": "[aadClientID]",
      "EncryptionOperation": "[encryptionOperation]",
      "KeyEncryptionAlgorithm": "[keyEncryptionAlgorithm]",
      "KeyVaultURL": "[keyVaultURL]",
      "KeyVaultResourceId": "[keyVaultResourceID]",
      "KeyEncryptionKeyURL": "[keyEncryptionKeyURL]",
      "KekVaultResourceId": "[kekVaultResourceID]",
      "SequenceVersion": "sequenceVersion",
      "VolumeType": "[volumeType]"
    }
  }
}
```

## Property values

Note: All values are case sensitive.

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2019-07-01	date
publisher	Microsoft.Azure.Security	string
type	AzureDiskEncryption	string
typeHandlerVersion	2.2, 1.1	string
(1.1 schema) AADClientID	xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx	guid
(1.1 schema) AADClientSecret	password	string
(1.1 schema) AADClientCertificate	thumbprint	string
EncryptionOperation	EnableEncryption	string
(optional - default RSA-OAEP ) KeyEncryptionAlgorithm	'RSA-OAEP', 'RSA-OAEP-256', 'RSA1_5'	string
KeyVaultURL	url	string
KeyVaultResourceId	url	string
(optional) KeyEncryptionKeyURL	url	string
(optional) KekVaultResourceId	url	string
(optional) SequenceVersion	uniqueidentifier	string
VolumeType	OS, Data, All	string

## Template deployment

For an example of template deployment based on schema v2.2, see Azure Quickstart Template [encrypt-running-windows-vm-without-aad](#).

For an example of template deployment based on schema v1.1, see Azure Quickstart Template [encrypt-running-windows-vm](#).

### NOTE

Also if `VolumeType` parameter is set to All, data disks will be encrypted only if they are properly formatted.

## Troubleshoot and support

### Troubleshoot

For troubleshooting, refer to the [Azure Disk Encryption troubleshooting guide](#).

### Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and](#)

[Stack Overflow forums](#).

Alternatively, you can file an Azure support incident. Go to [Azure support](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure Support FAQ](#).

## Next steps

- For more information about extensions, see [Virtual machine extensions and features for Windows](#).
- For more information about Azure Disk Encryption for Windows, see [Windows virtual machines](#).

# Key Vault virtual machine extension for Linux

9/21/2022 • 8 minutes to read • [Edit Online](#)

The Key Vault VM extension provides automatic refresh of certificates stored in an Azure key vault. Specifically, the extension monitors a list of observed certificates stored in key vaults. The extension retrieves and installs the corresponding certificates after detecting a change. The Key Vault VM extension is published and supported by Microsoft, currently on Linux VMs. This document details the supported platforms, configurations, and deployment options for the Key Vault VM extension for Linux.

## Operating system

The Key Vault VM extension supports these Linux distributions:

- Ubuntu-1804
- Suse-15
- [CBL-Mariner](#)

### NOTE

To get extended security features, prepare to upgrade Ubuntu-1604 and Debian-9 systems as these versions are reaching their end of designated support period.

### NOTE

The Key Vault VM Extension downloads the certificates in the default location or to the location provided by "certStoreLocation" property in the VM Extension settings. The KeyVault VM Extension updates the folder permission to 700 (drwx-----) allowing read, write and execute permission to the owner of the folder only

## Supported certificate content types

- PKCS #12
- PEM

## Prerequisites

- Key Vault instance with certificate. See [Create a Key Vault](#)
- VM/VMSS must have assigned [managed identity](#)
- The Key Vault Access Policy must be set with secrets `get` and `list` permission for VM/VMSS managed identity to retrieve a secret's portion of certificate. See [How to Authenticate to Key Vault](#) and [Assign a Key Vault access policy](#).
- VMSS should have the following identity setting:  

```
"identity": { "type": "UserAssigned", "userAssignedIdentities": { "[parameters('userAssignedIdentityResourceId')]: {} } }
```
- AKV extension should have this setting:  

```
"authenticationSettings": { "msiEndpoint": "[parameters('userAssignedIdentityEndpoint')]", "msiClientId": "[reference(parameters('userAssignedIdentityResourceId'), variables('msiApiVersion')).clientId]" }
```

## Key Vault VM extension version

- Ubuntu-18.04 and SUSE-15 users can chose to upgrade their key vault vm extension version to `v2.0` to avail full certificate chain download feature. Issuer certificates (intermediate and root) will be appended to the leaf certificate in the PEM file.
- If you prefer to upgrade to `v2.0`, you would need to delete `v1.0` first, then install `v2.0`.

```
az vm extension delete --name KeyVaultForLinux --resource-group ${resourceGroup} --vm-name ${vmName}
az vm extension set -n "KeyVaultForLinux" --publisher Microsoft.Azure.KeyVault --resource-group
"${resourceGroup}" --vm-name "${vmName}" -settings .\akvvm.json --version 2.0
```

The flag `--version 2.0` is optional because the latest version will be installed by default.

- If the VM has certificates downloaded by v1.0, deleting the v1.0 AKVVM extension will NOT delete the downloaded certificates. After installing v2.0, the existing certificates will NOT be modified. You would need to delete the certificate files or roll-over the certificate to get the PEM file with full-chain on the VM.

## Extension schema

The following JSON shows the schema for the Key Vault VM extension. The extension does not require protected settings - all its settings are considered information without security impact. The extension requires a list of monitored secrets, polling frequency, and the destination certificate store. Specifically:

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "KVVMExtensionForLinux",
  "apiVersion": "2019-07-01",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <vmName>)]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.KeyVault",
    "type": "KeyVaultForLinux",
    "typeHandlerVersion": "2.0",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "secretsManagementSettings": {
        "pollingIntervalInS": <polling interval in seconds, e.g. "3600">,
        "certificateStoreName": <It is ignored on Linux>,
        "linkOnRenewal": <Not available on Linux e.g.: false>,
        "certificateStoreLocation": <disk path where certificate is stored, default:
"/var/lib/waagent/Microsoft.Azure.KeyVault">,
        "requireInitialSync": <initial synchronization of certificates e..g: true>,
        "observedCertificates": <list of KeyVault URIs representing monitored certificates, e.g.:
["https://myvault.vault.azure.net/secrets/mycertificate",
"https://myvault.vault.azure.net/secrets/mycertificate2"]>
      },
      "authenticationSettings": {
        "msiEndpoint": <Optional MSI endpoint e.g.: "http://169.254.169.254/metadata/identity">,
        "msiClientId": <Optional MSI identity e.g.: "c7373ae5-91c2-4165-8ab6-7381d6e75619">
      }
    }
  }
}
```

**NOTE**

Your observed certificates URLs should be of the form <https://myVaultName.vault.azure.net/secrets/myCertName>.

This is because the `/secrets` path returns the full certificate, including the private key, while the `/certificates` path does not. More information about certificates can be found here: [Key Vault Certificates](#)

**IMPORTANT**

The 'authenticationSettings' property is required for VMs with **user assigned identities**. Set `msiClientId` to the identity that will authenticate to Key Vault.

Also required for **Azure Arc-enabled VMs**. Set `msiEndpoint` to <http://localhost:40342/metadata/identity>.

**Property values**

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2019-07-01	date
publisher	Microsoft.Azure.KeyVault	string
type	KeyVaultForLinux	string
typeHandlerVersion	2.0	int
pollingIntervalInS	3600	string
certificateStoreName	It is ignored on Linux	string
linkOnRenewal	false	boolean
certificateStoreLocation	/var/lib/waagent/Microsoft.Azure.KeyVault	string
requireInitialSync	true	boolean
observedCertificates	<code>["https://myvault.vault.azure.net/secrets/mycertificate", "https://myvault.vault.azure.net/secrets/mycertificate2"]</code>	string array
msiEndpoint	<a href="http://169.254.169.254/metadata/identity">http://169.254.169.254/metadata/identity</a>	string
msiClientId	c7373ae5-91c2-4165-8ab6-7381d6e75619	string

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment refresh of certificates. The extension can be deployed to individual VMs or virtual machine scale sets. The schema and configuration are common to both template types.

The JSON configuration for a virtual machine extension must be nested inside the virtual machine resource fragment of the template, specifically `"resources": []` object for the virtual machine template and for a virtual machine scale set under `"virtualMachineProfile": "extensionProfile": {"extensions" :[]}` object.

#### NOTE

The VM extension would require system or user managed identity to be assigned to authenticate to Key vault. See [How to authenticate to Key Vault and assign a Key Vault access policy](#).

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "KeyVaultForLinux",
  "apiVersion": "2019-07-01",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <vmName>)]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.KeyVault",
    "type": "KeyVaultForLinux",
    "typeHandlerVersion": "2.0",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "secretsManagementSettings": {
        "pollingIntervalInS": <polling interval in seconds, e.g. "3600">,
        "certificateStoreName": <ignored on linux>,
        "certificateStoreLocation": <disk path where certificate is stored, default: "/var/lib/waagent/Microsoft.Azure.KeyVault">,
        "observedCertificates": <list of KeyVault URIs representing monitored certificates, e.g.: "https://myvault.vault.azure.net/secrets/mycertificate"
      }
    }
  }
}
```

#### Extension Dependency Ordering

The Key Vault VM extension supports extension ordering if configured. By default the extension reports that it has successfully started as soon as it has started polling. However, it can be configured to wait until it has successfully downloaded the complete list of certificates before reporting a successful start. If other extensions depend on having the full set of certificates installed before they start, then enabling this setting will allow those extensions to declare a dependency on the Key Vault extension. This will prevent those extensions from starting until all certificates they depend on have been installed. The extension will retry the initial download indefinitely and remain in a `Transitioning` state.

To turn on extension dependency, set the following:

```
"secretsManagementSettings": {
  "requireInitialSync": true,
  ...
}
```

[Note] Using this feature is not compatible with an ARM template that creates a system assigned identity and updates a Key Vault access policy with that identity. Doing so will result in a deadlock as the vault access policy cannot be updated until all extensions have started. You should instead use a *single user assigned MSI identity* and pre-ACL your vaults with that identity before deploying.

# Azure PowerShell deployment

## WARNING

PowerShell clients often add `\` to `"` in the settings.json which will cause akvvm\_service fails with error:

```
[CertificateManagementConfiguration] Failed to parse the configuration settings with:not an object.
```

The Azure PowerShell can be used to deploy the Key Vault VM extension to an existing virtual machine or virtual machine scale set.

- To deploy the extension on a VM:

```
# Build settings
$settings = '{"secretsManagementSettings":'
{ "pollingIntervalInS": "' + <pollingInterval> +
'", "certificateStoreName": "' + <certStoreName> +
'", "certificateStoreLocation": "' + <certStoreLoc> +
'", "observedCertificates": ["' + <observedCert1> + '", "' + <observedCert2> + '"] } }'
$extName = "KeyVaultForLinux"
$extPublisher = "Microsoft.Azure.KeyVault"
$extType = "KeyVaultForLinux"

# Start the deployment
Set-AzVmExtension -TypeHandlerVersion "2.0" -ResourceGroupName <ResourceGroupName> -Location
<Location> -VMName <VMName> -Name $extName -Publisher $extPublisher -Type $extType -SettingString
$settings
```

- To deploy the extension on a virtual machine scale set:

```
# Build settings
$settings = '{"secretsManagementSettings":'
{ "pollingIntervalInS": "' + <pollingInterval> +
'", "certificateStoreName": "' + <certStoreName> +
'", "certificateStoreLocation": "' + <certStoreLoc> +
'", "observedCertificates": ["' + <observedCert1> + '", "' + <observedCert2> + '"] } }'
$extName = "KeyVaultForLinux"
$extPublisher = "Microsoft.Azure.KeyVault"
$extType = "KeyVaultForLinux"

# Add Extension to VMSS
$vmss = Get-AzVmss -ResourceGroupName <ResourceGroupName> -VMScaleSetName <VmssName>
Add-AzVmssExtension -VirtualMachineScaleSet $vmss -Name $extName -Publisher $extPublisher -Type
$extType -TypeHandlerVersion "2.0" -Setting $settings

# Start the deployment
Update-AzVmss -ResourceGroupName <ResourceGroupName> -VMScaleSetName <VmssName> -
VirtualMachineScaleSet $vmss
```

# Azure CLI deployment

The Azure CLI can be used to deploy the Key Vault VM extension to an existing virtual machine or virtual machine scale set.

- To deploy the extension on a VM:

```
# Start the deployment
az vm extension set -n "KeyVaultForLinux" ` 
--publisher Microsoft.Azure.KeyVault ` 
-g "<resourcegroup>" ` 
--vm-name "<vmName>" ` 
--version 2.0 ` 
--settings '{\"secretsManagementSettings\": { \"pollingIntervalInS\": \"<pollingInterval>\", 
\"certificateStoreName\": \"<certStoreName>\", \"certificateStoreLocation\": \"<certStoreLoc>\", 
\"observedCertificates\": [\" <observedCert1> \", \" <observedCert2> \"] }}'
```

- To deploy the extension on a virtual machine scale set:

```
# Start the deployment
az vmss extension set -n "KeyVaultForLinux" ` 
--publisher Microsoft.Azure.KeyVault ` 
-g "<resourcegroup>" ` 
--vmss-name "<vmssName>" ` 
--version 2.0 ` 
--settings '{\"secretsManagementSettings\": { \"pollingIntervalInS\": \"<pollingInterval>\", 
\"certificateStoreName\": \"<certStoreName>\", \"certificateStoreLocation\": \"<certStoreLoc>\", 
\"observedCertificates\": [\" <observedCert1> \", \" <observedCert2> \"] }}'
```

Please be aware of the following restrictions/requirements:

- Key Vault restrictions:
  - It must exist at the time of the deployment
  - The Key Vault Access Policy must be set for VM/VMSS Identity using a Managed Identity. See [How to Authenticate to Key Vault](#) and [Assign a Key Vault access policy](#).

## Troubleshoot and Support

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure PowerShell. To see the deployment state of extensions for a given VM, run the following command using the Azure PowerShell.

### Azure PowerShell

```
Get-AzVMExtension -VMName <vmName> -ResourceGroupName <resource group name>
```

### Azure CLI

```
az vm get-instance-view --resource-group <resource group name> --name <vmName> --query 
"instanceView.extensions"
```

The Azure CLI can run in several shell environments, but with slight format variations. If you have unexpected results with Azure CLI commands, see [How to use the Azure CLI successfully](#).

### Logs and configuration

The Key Vault VM extension logs only exist locally on the VM and are most informative when it comes to troubleshooting.

LOCATION	DESCRIPTION
/var/log/waagent.log	Shows when an update to the extension occurred.

LOCATION	DESCRIPTION
/var/log/azure/Microsoft.Azure.KeyVault.KeyVaultForLinux/*	Examine the Key Vault VM Extension service logs to determine the status of the akvvm_service service and certificate download. The download location of PEM files are also found in these files with an entry called certificate file name. If certificateStoreLocation is not specified, it will default to /var/lib/waagent/Microsoft.Azure.KeyVault.Store/
/var/lib/waagent/Microsoft.Azure.KeyVault.KeyVaultForLinux-<most recent version>/config/*	The configuration and binaries for Key Vault VM Extension service.

## Using Symlink

Symbolic links or Symlinks are advanced shortcuts. To avoid monitoring the folder and to get the latest certificate automatically, you can use this symlink `([VaultName].[CertificateName])` to get the latest version of certificate on Linux.

## Frequently Asked Questions

- Is there a limit on the number of observedCertificates you can setup? No, Key Vault VM Extension doesn't have limit on the number of observedCertificates.

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Key Vault virtual machine extension for Windows

9/21/2022 • 8 minutes to read • [Edit Online](#)

The Key Vault VM extension provides automatic refresh of certificates stored in an Azure key vault. Specifically, the extension monitors a list of observed certificates stored in key vaults, and, upon detecting a change, retrieves, and installs the corresponding certificates. This document details the supported platforms, configurations, and deployment options for the Key Vault VM extension for Windows.

## Operating system

The Key Vault VM extension supports below versions of Windows:

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012

The Key Vault VM extension is also supported on custom local VM that is uploaded and converted into a specialized image for use in Azure using Windows Server 2019 core install.

### NOTE

The Key Vault VM extension downloads all the certificates in the windows certificate store or to the location provided by "certificateStoreLocation" property in the VM extension settings. Currently, the KV VM extension grants access to the private key of the certificate only to the local system admin account. Additionally, it is currently not possible to define certificate store location per certificate. The VM extension team is working on a solution to close this feature gap.

## Supported certificate content types

- PKCS #12
- PEM

## Prerequisites

- Key Vault instance with certificate. See [Create a Key Vault](#)
- VM must have assigned [managed identity](#)
- The Key Vault Access Policy must be set with secrets `get` and `list` permission for VM/VMSS managed identity to retrieve a secret's portion of certificate. See [How to Authenticate to Key Vault](#) and [Assign a Key Vault access policy](#).
- Virtual Machine Scale Sets should have the following identity setting:

```
"identity": {  
    "type": "UserAssigned",  
    "userAssignedIdentities": {  
        "[parameters('userAssignedIdentityResourceId'))": {}  
    }  
}
```

- AKV extension should have this setting:

```

"authenticationSettings": {
    "msiEndpoint": "[parameters('userAssignedIdentityEndpoint')]",
    "msiClientId": "[reference(parameters('userAssignedIdentityResourceId'), variables('msiApiVersion')).clientId]"
}

```

## Extension schema

The following JSON shows the schema for the Key Vault VM extension. The extension does not require protected settings - all its settings are considered public information. The extension requires a list of monitored certificates, polling frequency, and the destination certificate store. Specifically:

```

{
    "type": "Microsoft.Compute/virtualMachines/extensions",
    "name": "KVVMExtensionForWindows",
    "apiVersion": "2019-07-01",
    "location": "<location>",
    "dependsOn": [
        "[concat('Microsoft.Compute/virtualMachines/', <vmName>)]"
    ],
    "properties": {
        "publisher": "Microsoft.Azure.KeyVault",
        "type": "KeyVaultForWindows",
        "typeHandlerVersion": "1.0",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "secretsManagementSettings": {
                "pollingIntervalInS": <string specifying polling interval in seconds, e.g: "3600">,
                "certificateStoreName": <certificate store name, e.g.: "MY">,
                "linkOnRenewal": <Only Windows. This feature ensures s-channel binding when certificate renews, without necessitating a re-deployment. e.g.: false>,
                "certificateStoreLocation": <certificate store location, currently it works locally only e.g.: "LocalMachine">,
                "requireInitialSync": <initial synchronization of certificates e.g: true>,
                "observedCertificates": <list of KeyVault URIs representing monitored certificates, e.g.: "https://myvault.vault.azure.net/secrets/mycertificate"
            },
            "authenticationSettings": {
                "msiEndpoint": <Optional MSI endpoint e.g.: "http://169.254.169.254/metadata/identity">,
                "msiClientId": <Optional MSI identity e.g.: "c7373ae5-91c2-4165-8ab6-7381d6e75619">
            }
        }
    }
}

```

### NOTE

Your observed certificates URLs should be of the form `https://myVaultName.vault.azure.net/secrets/myCertName`.

This is because the `/secrets` path returns the full certificate, including the private key, while the `/certificates` path does not. More information about certificates can be found here: [Key Vault Certificates](#)

### IMPORTANT

The 'authenticationSettings' property is required only for VMs with user assigned identities. It specifies identity to use for authentication to Key Vault.

**IMPORTANT**

If you specify the 'msiClientId', then the 'msiEndpoint' property is **required**. Usually the value should be set to  
<http://169.254.169.254/metadata/identity/oauth2/token>.

**Property values**

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2019-07-01	date
publisher	Microsoft.Azure.KeyVault	string
type	KeyVaultForWindows	string
typeHandlerVersion	1.0	int
pollingIntervalInS	3600	string
certificateStoreName	MY	string
linkOnRenewal	false	boolean
certificateStoreLocation	LocalMachine or CurrentUser (case sensitive)	string
requireInitialSync	true	boolean
observedCertificates	["https://myvault.vault.azure.net/secrets/mycertificate", "https://myvault.vault.azure.net/secrets/mycertificate2"]	string array
msiEndpoint	<a href="http://169.254.169.254/metadata/identity">http://169.254.169.254/metadata/identity</a>	string
msiClientId	c7373ae5-91c2-4165-8ab6-7381d6e75619	string

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment refresh of certificates. The extension can be deployed to individual VMs or virtual machine scale sets. The schema and configuration are common to both template types.

The JSON configuration for a virtual machine extension must be nested inside the virtual machine resource fragment of the template, specifically `"resources": []` object for the virtual machine template and in case of virtual machine scale set under `"virtualMachineProfile": "extensionProfile": {"extensions": []}` object.

#### NOTE

The VM extension would require system or user managed identity to be assigned to authenticate to Key vault. See [How to authenticate to Key Vault and assign a Key Vault access policy](#).

```
{  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "name": "KeyVaultForWindows",  
    "apiVersion": "2019-07-01",  
    "location": "<location>",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', <vmName>)]"  
    ],  
    "properties": {  
        "publisher": "Microsoft.Azure.KeyVault",  
        "type": "KeyVaultForWindows",  
        "typeHandlerVersion": "1.0",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "secretsManagementSettings": {  
                "pollingIntervalInS": <string specifying polling interval in seconds, e.g: "3600">,  
                "certificateStoreName": <certificate store name, e.g.: "MY">,  
                "certificateStoreLocation": <certificate store location, currently it works locally only e.g.:  
                "LocalMachine">,  
                "observedCertificates": <list of KeyVault URIs representing monitored certificates, e.g.:  
                ["https://myvault.vault.azure.net/secrets/mycertificate",  
                "https://myvault.vault.azure.net/secrets/mycertificate2"]>  
            }  
        }  
    }  
}
```

#### Extension Dependency Ordering

The Key Vault VM extension supports extension ordering if configured. By default the extension reports that it has successfully started as soon as it has started polling. However, it can be configured to wait until it has successfully downloaded the complete list of certificates before reporting a successful start. If other extensions depend on having the full set of certificates installed before they start, then enabling this setting will allow those extensions to declare a dependency on the Key Vault extension. This will prevent those extensions from starting until all certificates they depend on have been installed. The extension will retry the initial download indefinitely and remain in a `Transitioning` state.

To turn this on set the following:

```
"secretsManagementSettings": {  
    "requireInitialSync": true,  
    ...  
}
```

#### NOTE

Using this feature is not compatible with an ARM template that creates a system assigned identity and updates a Key Vault access policy with that identity. Doing so will result in a deadlock as the vault access policy cannot be updated until all extensions have started. You should instead use a *single user assigned MSI identity* and pre-ACL your vaults with that identity before deploying.

## Azure PowerShell deployment

## WARNING

PowerShell clients often add \ to " in the settings.json, which causes akvvm\_service to fail with the error

[CertificateManagementConfiguration] Failed to parse the configuration settings with: not an object. The extra \ and " characters will be visible in the portal, in Extensions under Settings. To avoid this, initialize \$settings as a PowerShell HashTable :

```
$settings = @{
    "secretsManagementSettings" = @{
        "pollingIntervalInS"      = "<pollingInterval>";
        "certificateStoreName"   = "<certStoreName>";
        "certificateStoreLocation" = "<certStoreLoc>";
        "observedCertificates"   = @("<observedCert1>", "<observedCert2>") } }
```

The Azure PowerShell can be used to deploy the Key Vault VM extension to an existing virtual machine or virtual machine scale set.

- To deploy the extension on a VM:

```
# Build settings
$settings = '{"secretsManagementSettings":'
{ "pollingIntervalInS": "' + <pollingInterval> +
'', "certificateStoreName": "' + <certStoreName> +
'', "certificateStoreLocation": "' + <certStoreLoc> +
'', "observedCertificates": ["' + <observedCert1> + '", "' + <observedCert2> + '"] } }'
$extName = "KeyVaultForWindows"
$extPublisher = "Microsoft.Azure.KeyVault"
$extType = "KeyVaultForWindows"

# Start the deployment
Set-AzVmExtension -TypeHandlerVersion "1.0" -ResourceGroupName <ResourceGroupName> -Location
<Location> -VMName <VMName> -Name $extName -Publisher $extPublisher -Type $extType -SettingString
$settings
```

- To deploy the extension on a virtual machine scale set:

```
# Build settings
$settings = '{"secretsManagementSettings":'
{ "pollingIntervalInS": "' + <pollingInterval> +
'', "certificateStoreName": "' + <certStoreName> +
'', "certificateStoreLocation": "' + <certStoreLoc> +
'', "observedCertificates": ["' + <observedCert1> + '", "' + <observedCert2> + '"] } }'
$extName = "KeyVaultForWindows"
$extPublisher = "Microsoft.Azure.KeyVault"
$extType = "KeyVaultForWindows"

# Add Extension to VMSS
$vmss = Get-AzVmss -ResourceGroupName <ResourceGroupName> -VMScaleSetName <VmssName>
Add-AzVmssExtension -VirtualMachineScaleSet $vmss -Name $extName -Publisher $extPublisher -Type
$extType -TypeHandlerVersion "1.0" -Setting $settings

# Start the deployment
Update-AzVmss -ResourceGroupName <ResourceGroupName> -VMScaleSetName <VmssName> -
VirtualMachineScaleSet $vmss
```

# Azure CLI deployment

The Azure CLI can be used to deploy the Key Vault VM extension to an existing virtual machine or virtual machine scale set.

- To deploy the extension on a VM:

```
# Start the deployment
az vm extension set --name "KeyVaultForWindows" ` 
--publisher Microsoft.Azure.KeyVault ` 
--resource-group "<resourcegroup>" ` 
--vm-name "<vmName>" ` 
--settings '{\"secretsManagementSettings\": { \"pollingIntervalInS\": \"<pollingInterval>\", 
\"certificateStoreName\": \"<certStoreName>\", \"certificateStoreLocation\": \"<certStoreLoc>\", 
\"observedCertificates\": [\" <observedCert1> \", \" <observedCert2> \"] }}'
```

- To deploy the extension on a virtual machine scale set :

```
# Start the deployment
az vmss extension set --name "KeyVaultForWindows" ` 
--publisher Microsoft.Azure.KeyVault ` 
--resource-group "<resourcegroup>" ` 
--vmss-name "<vmName>" ` 
--settings '{\"secretsManagementSettings\": { \"pollingIntervalInS\": \"<pollingInterval>\", 
\"certificateStoreName\": \"<certStoreName>\", \"certificateStoreLocation\": \"<certStoreLoc>\", 
\"observedCertificates\": [\" <observedCert1> \", \" <observedCert2> \"] }}'
```

Please be aware of the following restrictions/requirements:

- Key Vault restrictions:
  - It must exist at the time of the deployment
  - The Key Vault Access Policy must be set for VM/VMSS Identity using a Managed Identity. See [How to Authenticate to Key Vault](#) and [Assign a Key Vault access policy](#).

## Troubleshoot and support

### Frequently Asked Questions

- Is there a limit on the number of observedCertificates you can setup? No, Key Vault VM Extension doesn't have limit on the number of observedCertificates.

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure PowerShell. To see the deployment state of extensions for a given VM, run the following command using the Azure PowerShell.

### Azure PowerShell

```
Get-AzVMExtension -VMName <vmName> -ResourceGroupName <resource group name>
```

### Azure CLI

```
az vm get-instance-view --resource-group <resource group name> --name <vmName> --query 
"instanceView.extensions"
```

### Logs and configuration

The Key Vault VM extension logs only exist locally on the VM and are most informative when it comes to troubleshooting

LOCATION	DESCRIPTION
C:\WindowsAzure\Logs\WaAppAgent.log	Shows when an update to the extension occurred.
C:\WindowsAzure\Logs\Plugins\Microsoft.Azure.KeyVault.KeyVaultForWindows\<most recent version>\	Shows the status of certificate download. The download location will always be the Windows computer's MY store (certlm.msc).
C:\Packages\Plugins\Microsoft.Azure.KeyVault.KeyVaultForWindows\<most recent version>\RuntimeSettings\	The Key Vault VM Extension service logs show the status of the akvvm_service service.
C:\Packages\Plugins\Microsoft.Azure.KeyVault.KeyVaultForWindows\<most recent version>\Status\	The configuration and binaries for Key Vault VM Extension service.

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Overview of the guest configuration extension

9/21/2022 • 4 minutes to read • [Edit Online](#)

The Guest Configuration extension is a component of Azure Policy that performs audit and configuration operations inside virtual machines. Policies such as security baseline definitions for [Linux](#) and [Windows](#) can't check settings inside machines until the extension is installed.

## Prerequisites

For the machine to authenticate to the Guest Configuration service, the machine must have a [System-Assigned Managed Identity](#). The identity requirement on a virtual machine is met if the following property is set.

```
"identity": {  
    "type": "SystemAssigned"  
}
```

### Operating Systems

Support for the Guest Configuration extension is the same as operating system support [documented for the end to end solution](#).

### Internet connectivity

The agent installed by the Guest Configuration extension must be able to reach content packages listed by Guest Configuration assignments, and report status to the Guest Configuration service. The machine can connect using outbound HTTPS over TCP port 443, or if a connection is provided through private networking. To learn more about private networking, see the following articles:

- [Guest Configuration, communicate over private link in Azure](#)
- [Use private endpoints for Azure Storage](#)

## How can I install the extension?

The instance name of the extension must be set to "AzurePolicyforWindows" or "AzurePolicyforLinux", because the policies referenced above require these specific strings.

By default, all deployments update to the latest version. The value of property *autoUpgradeMinorVersion* defaults to "true" unless it is otherwise specified. You do not need to worry about updating your code when new versions of the extension are released.

## Automatic upgrade

The guest configuration extension supports property `enableAutomaticUpgrade`. When this property is set to `true`, Azure will automatically upgrade to the latest version of the extension as future releases become available. For more information, see the page [Automatic Extension Upgrade for VMs and Scale Sets in Azure](#)

### Azure Policy

To deploy the latest version of the extension at scale including identity requirements, [assign](#) the Azure Policy:

[Deploy prerequisites to enable Guest Configuration policies on virtual machines](#).

### Azure CLI

To deploy the extension for Linux:

```
az vm extension set --publisher Microsoft.GuestConfiguration --name ConfigurationforLinux --extension-instance-name AzurePolicyforLinux --resource-group myResourceGroup --vm-name myVM --enable-auto-upgrade true
```

To deploy the extension for Windows:

```
az vm extension set --publisher Microsoft.GuestConfiguration --name ConfigurationforWindows --extension-instance-name AzurePolicyforWindows --resource-group myResourceGroup --vm-name myVM --enable-auto-upgrade true
```

## PowerShell

To deploy the extension for Linux:

```
Set-AzVMExtension -Publisher 'Microsoft.GuestConfiguration' -Type 'ConfigurationforLinux' -Name 'AzurePolicyforLinux' -TypeHandlerVersion 1.0 -ResourceGroupName 'myResourceGroup' -Location 'myLocation' -VMName 'myVM' -EnableAutomaticUpgrade $true
```

To deploy the extension for Windows:

```
Set-AzVMExtension -Publisher 'Microsoft.GuestConfiguration' -Type 'ConfigurationforWindows' -Name 'AzurePolicyforWindows' -TypeHandlerVersion 1.0 -ResourceGroupName 'myResourceGroup' -Location 'myLocation' -VMName 'myVM' -EnableAutomaticUpgrade $true
```

## Resource Manager template

To deploy the extension for Linux:

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "[concat(parameters('VMName'), '/AzurePolicyforLinux')]",
  "apiVersion": "2020-12-01",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', parameters('VMName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.GuestConfiguration",
    "type": "ConfigurationforLinux",
    "typeHandlerVersion": "1.0",
    "autoUpgradeMinorVersion": true,
    "enableAutomaticUpgrade": true,
    "settings": {},
    "protectedSettings": {}
  }
}
```

To deploy the extension for Windows:

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "[concat(parameters('VMName'), '/AzurePolicyforWindows')]",
  "apiVersion": "2020-12-01",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', parameters('VMName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.GuestConfiguration",
    "type": "ConfigurationforWindows",
    "typeHandlerVersion": "1.0",
    "autoUpgradeMinorVersion": true,
    "enableAutomaticUpgrade": true,
    "settings": {},
    "protectedSettings": {}
  }
}
```

## Bicep

To deploy the extension for Linux:

```
resource virtualMachine 'Microsoft.Compute/virtualMachines@2021-03-01' existing = {
  name: 'VMName'
}
resource windowsVMGuestConfigExtension 'Microsoft.Compute/virtualMachines/extensions@2020-12-01' = {
  parent: virtualMachine
  name: 'AzurePolicyforLinux'
  location: resourceGroup().location
  properties: {
    publisher: 'Microsoft.GuestConfiguration'
    type: 'ConfigurationforLinux'
    typeHandlerVersion: '1.0'
    autoUpgradeMinorVersion: true
    enableAutomaticUpgrade: true
    settings: {}
    protectedSettings: {}
  }
}
```

To deploy the extension for Windows:

```
resource virtualMachine 'Microsoft.Compute/virtualMachines@2021-03-01' existing = {
  name: 'VMName'
}
resource windowsVMGuestConfigExtension 'Microsoft.Compute/virtualMachines/extensions@2020-12-01' = {
  parent: virtualMachine
  name: 'AzurePolicyforWindows'
  location: resourceGroup().location
  properties: {
    publisher: 'Microsoft.GuestConfiguration'
    type: 'ConfigurationforWindows'
    typeHandlerVersion: '1.0'
    autoUpgradeMinorVersion: true
    enableAutomaticUpgrade: true
    settings: {}
    protectedSettings: {}
  }
}
```

## Terraform

To deploy the extension for Linux:

```
resource "azurerm_virtual_machine_extension" "gc" {
    name          = "AzurePolicyforLinux"
    virtual_machine_id = "myVMID"
    publisher     = "Microsoft.GuestConfiguration"
    type          = "ConfigurationforLinux"
    type_handler_version = "1.0"
    auto_upgrade_minor_version = "true"
}
```

To deploy the extension for Windows:

```
resource "azurerm_virtual_machine_extension" "gc" {
    name          = "AzurePolicyforWindows"
    virtual_machine_id = "myVMID"
    publisher     = "Microsoft.GuestConfiguration"
    type          = "ConfigurationforWindows"
    type_handler_version = "1.0"
    auto_upgrade_minor_version = "true"
}
```

## Settings

There's no need to include any settings or protected-settings properties on the extension. All such information is retrieved by the agent from [Guest Configuration assignment](#) resources. For example, the [ConfigurationUri](#), [Mode](#), and [ConfigurationSetting](#) properties are each managed per-configuration rather than on the VM extension.

## Guest Configuration resource provider error codes

See below for a list of the possible error messages when enabling the extension

ERROR CODE	DESCRIPTION
NoComplianceReport	VM has not reported the compliance data.
GCExtensionMissing	Guest Configuration extension is missing.
ManagedIdentityMissing	Managed identity is missing.
UserIdentityMissing	User assigned identity is missing.
GCExtensionManagedIdentityMissing	Guest Configuration extension and managed identity is missing.
GCExtensionUserIdentityMissing	Guest Configuration extension and user identity is missing.
GCExtensionIdentityMissing	Guest Configuration extension, managed identity and user identity are missing.

## Next steps

- For more information about Azure Policy's guest configuration, see [Understand Azure Policy's Guest Configuration](#)

- For more information about how the Linux Agent and extensions work, see [Azure VM extensions and features for Linux](#).
- For more information about how the Windows Guest Agent and extensions work, see [Azure VM extensions and features for Windows](#).
- To install the Windows Guest Agent, see [Azure Windows Virtual Machine Agent Overview](#).
- To install the Linux Agent, see [Azure Linux Virtual Machine Agent Overview](#).

# Use the Azure Custom Script Extension Version 2 with Linux virtual machines

9/21/2022 • 13 minutes to read • [Edit Online](#)

The Custom Script Extension Version 2 downloads and runs scripts on Azure virtual machines (VMs). This extension is useful for post-deployment configuration, software installation, or any other configuration or management task. You can download scripts from Azure Storage or another accessible internet location, or you can provide them to the extension runtime.

The Custom Script Extension integrates with Azure Resource Manager templates. You can also run it by using the Azure CLI, PowerShell, or the Azure Virtual Machines REST API.

This article details how to use the Custom Script Extension from the Azure CLI, and how to run the extension by using an Azure Resource Manager template. This article also provides troubleshooting steps for Linux systems.

There are two Linux Custom Script Extensions:

- Version 1: Microsoft.OSTCExtensions.CustomScriptForLinux
- Version 2: Microsoft.Azure.Extensions.CustomScript

Please switch new and existing deployments to use Version 2. The new version is a drop-in replacement. The migration is as easy as changing the name and version. You don't need to change your extension configuration.

## Prerequisites

### Operating system

The Custom Script Extension for Linux will run on supported operating systems. For more information, see [Endorsed Linux distributions on Azure](#).

### Script location

You can set the extension to use your Azure Blob Storage credentials so that it can access Azure Blob Storage. The script location can be anywhere, as long as the VM can route to that endpoint (for example, GitHub or an internal file server).

### Internet connectivity

If you need to download a script externally, such as from GitHub or Azure Storage, then you need to open additional firewall or network security group (NSG) ports. For example, if your script is located in Azure Storage, you can allow access by using Azure NSG [service tags for Storage](#).

If your script is on a local server, you might still need to open additional firewall or NSG ports.

### Tips and tricks

- The highest failure rate for this extension is due to syntax errors in the script. Test that the script runs without errors. Put additional logging into the script to make it easier to find failures.
- Write scripts that are idempotent, so running them more than once accidentally won't cause system changes.
- Ensure that the scripts don't require user input when they run.
- The script is allowed 90 minutes to run. Anything longer will result in a failed provision of the extension.
- Don't put reboots inside the script. This action will cause problems with other extensions that are being installed, and the extension won't continue after the reboot.
- If you have a script that will cause a reboot before installing applications and running scripts, schedule the reboot by using a Cron job or by using tools such as DSC, Chef, or Puppet extensions.
- Don't run a script that will cause a stop or update of the VM agent. It might leave the extension in a transitioning state and lead to a timeout.

- The extension will run a script only once. If you want to run a script on every startup, you can use a [cloud-init image](#) and use a [Scripts Per Boot](#) module. Alternatively, you can use the script to create a [systemd](#) service unit.
- You can have only one version of an extension applied to the VM. To run a second custom script, you can update the existing extension with a new configuration. Alternatively, you can remove the custom script extension and reapply it with the updated script.
- If you want to schedule when a script will run, use the extension to create a Cron job.
- When the script is running, you'll only see a "transitioning" extension status from the Azure portal or CLI. If you want more frequent status updates for a running script, you'll need to create your own solution.
- The Custom Script Extension doesn't natively support proxy servers. However, you can use a file transfer tool that supports proxy servers within your script, such as [Curl](#).
- Be aware of non-default directory locations that your scripts or commands might rely on. Have logic to handle this situation.

## Extension schema

The Custom Script Extension configuration specifies things like script location and the command to be run. You can store this information in configuration files, specify it on the command line, or specify it in an Azure Resource Manager template.

You can store sensitive data in a protected configuration, which is encrypted and only decrypted on the target virtual machine. The protected configuration is useful when the execution command includes secrets such as a password. Here's an example:

```
{
  "name": "config-app",
  "type": "Extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "2019-03-01",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', concat(variables('vmName'),copyindex()))]"
  ],
  "tags": {
    "displayName": "config-app"
  },
  "properties": {
    "publisher": "Microsoft.Azure.Extensions",
    "type": "CustomScript",
    "typeHandlerVersion": "2.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "skipDOS2Unix":false,
      "timestamp":123456789
    },
    "protectedSettings": {
      "commandToExecute": "<command-to-execute>",
      "script": "<base64-script-to-execute>",
      "storageAccountName": "<storage-account-name>",
      "storageAccountKey": "<storage-account-key>",
      "fileUris": ["https://.."],
      "managedIdentity" : "<managed-identity-identifier>"
    }
  }
}
```

### NOTE

The `managedIdentity` property *must not* be used in conjunction with the `storageAccountName` or `storageAccountKey` property.

## Property values

NAME	VALUE OR EXAMPLE	DATA TYPE
apiVersion	2019-03-01	date
publisher	Microsoft.Azure.Extensions	string
type	CustomScript	string
typeHandlerVersion	2.1	int
fileUris	https://github.com/MyProject/Archive/myPythonScript.py	
commandToExecute	python MyPythonScript.py \<my-param>	string
script	IyEvYmluL3NoCmVjaG8gIlVwZGF0aW5nIHBhY2tld2VzIC4uLiIKYXB0IHVwZGF0ZQphcHQgdXBncmFkZSAte	
skipDos2Unix	false	Boolean
timestamp	123456789	32-bit integer
storageAccountName	examplestorageacct	string
storageAccountKey	TmJK/1N3AbAZ3q/+h0Xoi/173z0qsaxXDhq99\$UpXQp2DQIBuv2Tifp60cE/OaHsJZmQZ7teQfczQj8h	
managedIdentity	{ } or { "clientId": "31b403aa-c364-4240-a7ff-d85fb6cd7232" } or { "objectId": "12dd289c-0583-46e5-b9b4-115d5c19ef4b" }	JSON object

## Property value details

PROPERTY	OPTIONAL OR REQUIRED	DETAILS
apiVersion	Not applicable	You can find the most up-to-date API version by using <a href="#">Resource Explorer</a> or by using the command <code>az provider list -o json</code> in the Azure CLI.
fileUris	Optional	URLs for files to be downloaded.
commandToExecute	Required if <code>script</code> isn't set	The entry point script to run. Use this property instead of <code>script</code> if your command contains secrets such as passwords.
script	Required if <code>commandToExecute</code> isn't set	A Base64-encoded (and optionally gzip'ed) script run by <code>/bin/sh</code> .
skipDos2Unix	Optional	Set this value to <code>false</code> if you want to skip dos2unix conversion of script-based file URLs or scripts.

PROPERTY	OPTIONAL OR REQUIRED	DETAILS
<code>timestamp</code>	Optional	Change this value only to trigger a rerun of the script. Any integer value is acceptable, as long as it's different from the previous value.
<code>storageAccountName</code>	Optional	The name of storage account. If you specify storage credentials, all <code>fileUris</code> values must be URLs for Azure blobs.
<code>storageAccountKey</code>	Optional	The access key of the storage account.
<code>managedIdentity</code>	Optional	The <a href="#">managed identity</a> for downloading files:  <code>clientId</code> (optional, string): The client ID of the managed identity.  <code>objectId</code> (optional, string): The object ID of the managed identity.

You can set the following values in either public or protected settings. The extension will reject any configuration where these values are set in both public and protected settings.

- `commandToExecute`
- `script`
- `fileUris`

Using public settings might be useful for debugging, but we strongly recommend that you use protected settings.

Public settings are sent in clear text to the VM where the script will be run. Protected settings are encrypted through a key known only to Azure and the VM. The settings are saved to the VM as they were sent. That is, if the settings were encrypted, they're saved encrypted on the VM. The certificate that's used to decrypt the encrypted values is stored on the VM. The certificate is also used to decrypt settings (if necessary) at runtime.

#### Property: `skipDos2Unix`

The default value is `false`, which means dos2unix conversion *is* executed.

The previous version of the Custom Script Extension, Microsoft.OSTCExtensions.CustomScriptForLinux, would automatically convert DOS files to UNIX files by translating `\r\n` to `\n`. This translation still exists and is on by default. This conversion is applied to all files downloaded from `fileUris` or the script setting based on either of the following criteria:

- The extension is `.sh`, `.txt`, `.py`, or `.pl`. The script setting will always match this criterion because it's assumed to be a script run with `/bin/sh`. The script setting is saved as `script.sh` on the VM.
- The file starts with `#!`.

You can skip the dos2unix conversion by setting `skipDos2Unix` to `true`:

```
{
  "fileUris": ["<url>"],
  "commandToExecute": "<command-to-execute>",
  "skipDos2Unix": true
}
```

#### Property: `script`

The Custom Script Extension supports execution of a user-defined script. The script settings combine `commandToExecute` and `fileUris` into a single setting. Instead of having to set up a file for download from Azure

Storage or a GitHub gist, you can simply encode the script as a setting. You can use the script to replace `commandToExecute` and `fileUris`.

Here are some requirements:

- The script *must* be Base64 encoded.
- The script can *optionally* be gzip'ed.
- You can use the script setting in public or protected settings.
- The maximum size of the script parameter's data is 256 KB. If the script exceeds this size, it won't be run.

For example, the following script is saved to the file `/script.sh`:

```
#!/bin/sh
echo "Updating packages ..."
apt update
apt upgrade -y
```

You would construct the correct Custom Script Extension script setting by taking the output of the following command:

```
cat script.sh | base64 -w0
```

```
{
  "script": "IyEvYmluL3NoCmVjaG8gIlVwZGF0aW5nIHBhY2thZ2VzIC4uLiIKYXB0IHVwZGF0ZQphcHQgdXBncmFkZSAteQo="
}
```

In most cases, the script can optionally be gzip'ed to further reduce size. The Custom Script Extension automatically detects the use of gzip compression.

```
cat script | gzip -9 | base64 -w 0
```

The Custom Script Extension uses the following algorithm to run a script:

1. Assert that the length of the script's value does not exceed 256 KB.
2. Base64 decode the script's value.
3. Try to gunzip the Base64-decoded value.
4. Write the decoded (and optionally decompressed) value to disk (`/var/lib/waagent/custom-script/#/script.sh`).
5. Run the script by using `/bin/sh -c /var/lib/waagent/custom-script/#/script.sh`.

#### Property: managedIdentity

##### NOTE

This property *must* be specified in protected settings only.

The Custom Script Extension (version 2.1 and later) supports [managed identities](#) for downloading files from URLs provided in the `fileUris` setting. It allows the Custom Script Extension to access Azure Storage private blobs or containers without the user having to pass secrets like shared access signature (SAS) tokens or storage account keys.

To use this feature, the user must add a [system-assigned](#) or [user-assigned](#) identity to the VM or virtual machine scale set where the Custom Script Extension is expected to run. The user must then [grant the managed identity access to the Azure Storage container or blob](#).

To use the system-assigned identity on the target VM or virtual machine scale set, set `managedIdentity` to an empty JSON object.

Example:

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.sh"],  
    "commandToExecute": "sh script1.sh",  
    "managedIdentity" : {}  
}
```

To use the user-assigned identity on the target VM or virtual machine scale set, configure `managedIdentity` with the client ID or the object ID of the managed identity.

Examples:

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.sh"],  
    "commandToExecute": "sh script1.sh",  
    "managedIdentity" : { "clientId": "31b403aa-c364-4240-a7ff-d85fb6cd7232" }  
}
```

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.sh"],  
    "commandToExecute": "sh script1.sh",  
    "managedIdentity" : { "objectId": "12dd289c-0583-46e5-b9b4-115d5c19ef4b" }  
}
```

**NOTE**

The `managedIdentity` property *must not* be used in conjunction with the `storageAccountName` or `storageAccountKey` property.

## Template deployment

You can deploy Azure VM extensions by using Azure Resource Manager templates. The JSON schema detailed in the previous section can be used in an Azure Resource Manager template to run the Custom Script Extension during the template's deployment. You can find a sample template that includes the Custom Script Extension on [GitHub](#).

```
{
  "name": "config-app",
  "type": "extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "2019-03-01",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', concat(variables('vmName'),copyindex()))]"
  ],
  "tags": {
    "displayName": "config-app"
  },
  "properties": {
    "publisher": "Microsoft.Azure.Extensions",
    "type": "CustomScript",
    "typeHandlerVersion": "2.1",
    "autoUpgradeMinorVersion": true,
    "settings": {
    },
    "protectedSettings": {
      "commandToExecute": "sh hello.sh <param2>",
      "fileUris": ["https://github.com/MyProject/Archive/hello.sh"]
    }
  }
}
```

#### NOTE

These property names are case-sensitive. To avoid deployment problems, use the names as shown here.

## Azure CLI

When you're using the Azure CLI to run the Custom Script Extension, create a configuration file or files. At a minimum, you must have `commandToExecute`.

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM --name customScript \
--publisher Microsoft.Azure.Extensions \
--protected-settings ./script-config.json
```

Optionally, you can specify the settings in the command as a JSON-formatted string. This allows the configuration to be specified during execution and without a separate configuration file.

```
az vm extension set \
--resource-group exttest \
--vm-name exttest \
--name customScript \
--publisher Microsoft.Azure.Extensions \
--protected-settings '{"fileUris": ["https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-linux/scripts/config-music.sh"], "commandToExecute": "./config-music.sh"}'
```

#### Example: Public configuration with script file

```
{
  "fileUris": ["https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-linux/scripts/config-music.sh"],
  "commandToExecute": "./config-music.sh"
}
```

Azure CLI command:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM --name customScript \
--publisher Microsoft.Azure.Extensions \
--settings ./script-config.json
```

#### Example: Public configuration with no script file

```
{
  "commandToExecute": "apt-get -y update && apt-get install -y apache2"
}
```

Azure CLI command:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM --name customScript \
--publisher Microsoft.Azure.Extensions \
--settings ./script-config.json
```

#### Example: Public and protected configuration files

You use a public configuration file to specify the script file's URI. You use a protected configuration file to specify the command to be run.

Public configuration file:

```
{
  "fileUris": ["https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-linux/scripts/config-music.sh"]
}
```

Protected configuration file:

```
{
  "commandToExecute": "./config-music.sh <param1>"
}
```

Azure CLI command:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name customScript \
--publisher Microsoft.Azure.Extensions \
--settings ./script-config.json \
--protected-settings ./protected-config.json
```

## Virtual machine scale sets

If you deploy the Custom Script Extension from the Azure portal, you don't have control over the expiration of the SAS token for accessing the script in your storage account. The result is that the initial deployment works, but when the storage account's SAS token expires, any subsequent scaling operation fails because the Custom Script Extension can no longer access the storage account.

We recommend that you use [PowerShell](#), the [Azure CLI](#), or an [Azure Resource Manager template](#) when you deploy the Custom Script Extension on a virtual machine scale set. This way, you can choose to use a managed identity or have direct control of the expiration of the SAS token for accessing the script in your storage account for as long as you need.

## Troubleshooting

When the Custom Script Extension runs, the script is created or downloaded into a directory that's similar to the following example. The command output is also saved into this directory in `stdout` and `stderr` files.

```
/var/lib/waagent/custom-script/download/0/
```

To troubleshoot, first check the Linux Agent Log and ensure that the extension ran:

```
/var/log/waagent.log
```

Look for the extension execution. It will look something like:

```
2018/04/26 17:47:22.110231 INFO [Microsoft.Azure.Extensions.customScript-2.0.6] [Enable] current handler state is: notinstalled
2018/04/26 17:47:22.306407 INFO Event: name=Microsoft.Azure.Extensions.customScript, op=Download, message=Download succeeded, duration=167
2018/04/26 17:47:22.339958 INFO [Microsoft.Azure.Extensions.customScript-2.0.6] Initialize extension directory
2018/04/26 17:47:22.368293 INFO [Microsoft.Azure.Extensions.customScript-2.0.6] Update settings file: 0.settings
2018/04/26 17:47:22.394482 INFO [Microsoft.Azure.Extensions.customScript-2.0.6] Install extension [bin/custom-script-shim install]
2018/04/26 17:47:23.432774 INFO Event: name=Microsoft.Azure.Extensions.customScript, op=Install, message=Launch command succeeded: bin/custom-script-shim install, duration=1007
2018/04/26 17:47:23.476151 INFO [Microsoft.Azure.Extensions.customScript-2.0.6] Enable extension [bin/custom-script-shim enable]
2018/04/26 17:47:24.516444 INFO Event: name=Microsoft.Azure.Extensions.customScript, op=Enable, message=Launch command succeeded: bin/custom-sc
```

In the preceding output:

- `Enable` is when the command starts running.
- `Download` relates to the downloading of the Custom Script Extension package from Azure, not the script files specified in `fileUris`.

The Azure Script Extension produces a log, which you can find here:

```
/var/log/azure/custom-script/handler.log
```

Look for the individual execution. It will look something like:

```
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event=start
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event=pre-check
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="comparing seqnum"
path=mrseq
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="seqnum saved"
path=mrseq
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="reading configuration"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="read configuration"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="validating json schema"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="json schema valid"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="parsing configuration json"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="parsed configuration json"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="validating configuration logically"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="validated configuration"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="creating output directory" path=/var/lib/waagent/custom-script/download/0
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="created output directory"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 files=1
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 file=0 event="download start"
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 file=0 event="download complete" output=/var/lib/waagent/custom-script/download/0
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="executing command" output=/var/lib/waagent/custom-script/download/0
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="executing protected commandToExecute" output=/var/lib/waagent/custom-script/download/0
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event="executed command" output=/var/lib/waagent/custom-script/download/0
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event=enabled
time=2018-04-26T17:47:23Z version=v2.0.6/git@1008306-clean operation=enable seq=0 event=end
```

Here you can see:

- The `enable` command that starts this log.
- The settings passed to the extension.
- The extension downloading the file and the result of that.
- The command being run and the result.

You can also retrieve the execution state of the Custom Script Extension, including the actual arguments passed as `commandToExecute`, by using the Azure CLI:

```
az vm extension list -g myResourceGroup --vm-name myVM
```

The output looks like the following text:

```
[  
  {  
    "autoUpgradeMinorVersion": true,  
    "forceUpdateTag": null,  
    "id":  
      "/subscriptions/subscriptionid/resourceGroups/rgname/providers/Microsoft.Compute/virtualMachines/vmname/extensions/customscript",  
    "resourceGroup": "rgname",  
    "settings": {  
      "commandToExecute": "sh script.sh > ",  
      "fileUris": [  
        "https://catalogartifact.azureedge.net/publicartifacts/scripts/script.sh",  
        "https://catalogartifact.azureedge.net/publicartifacts/scripts/script.sh"  
      ]  
    },  
    "tags": null,  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "typeHandlerVersion": "2.0",  
    "virtualMachineExtensionType": "CustomScript"  
  },  
  {  
    "autoUpgradeMinorVersion": true,  
    "forceUpdateTag": null,  
    "id":  
      "/subscriptions/subscriptionid/resourceGroups/rgname/providers/Microsoft.Compute/virtualMachines/vmname/extensions/OmsAgentForLinux",  
    "instanceView": null,  
    "location": "eastus",  
    "name": "OmsAgentForLinux",  
    "protectedSettings": null,  
    "provisioningState": "Succeeded",  
    "publisher": "Microsoft.EnterpriseCloud.Monitoring",  
    "resourceGroup": "rgname",  
    "settings": {  
      "workspaceId": "workspaceid"  
    },  
    "tags": null,  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "typeHandlerVersion": "1.0",  
    "virtualMachineExtensionType": "OmsAgentForLinux"  
  }  
]
```

#### Azure CLI syntax issues

The Azure CLI can run in several shell environments, but with slight format variations. If you have unexpected results with Azure CLI commands, see [How to use the Azure CLI successfully](#).

## Next steps

To see the code, current issues, and versions, go to the [custom-script-extension-linux repo on GitHub](#).

# Custom Script Extension for Windows

9/21/2022 • 12 minutes to read • [Edit Online](#)

The Custom Script Extension downloads and runs scripts on Azure virtual machines (VMs). This extension is useful for post-deployment configuration, software installation, or any other configuration or management task. You can download scripts from Azure Storage or GitHub, or provide them to the Azure portal at extension runtime.

The Custom Script Extension integrates with Azure Resource Manager templates. You can also run it by using the Azure CLI, PowerShell, the Azure portal, or the Azure Virtual Machines REST API.

This article details how to use the Custom Script Extension by using the Azure PowerShell module and Azure Resource Manager templates. It also provides troubleshooting steps for Windows systems.

## Prerequisites

### NOTE

Don't use the Custom Script Extension to run `Update-AzVM` with the same VM as its parameter, because it will wait for itself.

### Operating system

The Custom Script Extension for Windows will run on these supported operating systems:

- Windows Server 2008 R2
- Windows Server 2012
- Windows Server 2012 R2
- Windows 10
- Windows Server 2016
- Windows Server 2016 Core
- Windows Server 2019
- Windows Server 2019 Core
- Windows Server 2022
- Windows Server 2022 Core
- Windows 11

### Script location

You can set the extension to use your Azure Blob Storage credentials so that it can access Azure Blob Storage. The script location can be anywhere, as long as the VM can route to that endpoint (for example, GitHub or an internal file server).

### Internet connectivity

If you need to download a script externally, such as from GitHub or Azure Storage, then you need to open additional firewall or network security group (NSG) ports. For example, if your script is located in Azure Storage, you can allow access by using Azure NSG [service tags for Storage](#).

The Custom Script Extension does not have any way to bypass certificate validation. So if you're downloading from a secured location with, for example, a self-signed certificate, you might get errors like "The remote certificate is invalid according to the validation procedure." Make sure that the certificate is correctly installed in the *Trusted Root Certification Authorities* store on the VM.

If your script is on a local server, you might still need to open additional firewall or NSG ports.

### Tips and tricks

- The highest failure rate for this extension is due to syntax errors in the script. Test that the script runs without errors. Put additional logging into the script to make it easier to find failures.
- Write scripts that are idempotent, so running them more than once accidentally won't cause system changes.
- Ensure that the scripts don't require user input when they run.
- The script is allowed 90 minutes to run. Anything longer will result in a failed provision of the extension.
- Don't put reboots inside the script. This action will cause problems with other extensions that are being installed, and the extension won't continue after the reboot.
- If you have a script that will cause a reboot before installing applications and running scripts, schedule the reboot by using a Windows Scheduled Task or by using tools such as DSC, Chef, or Puppet extensions.
- Don't run a script that will cause a stop or update of the VM agent. It might leave the extension in a transitioning state and lead to a timeout.
- The extension will run a script only once. If you want to run a script on every startup, use the extension to create a Windows Scheduled Task.
- If you want to schedule when a script will run, use the extension to create a Windows Scheduled Task.
- When the script is running, you'll only see a "transitioning" extension status from the Azure portal or CLI. If you want more frequent status updates for a running script, you'll need to create your own solution.
- The Custom Script Extension doesn't natively support proxy servers. However, you can use a file transfer tool

that supports proxy servers within your script, such as *Invoke-WebRequest*.

- Be aware of non-default directory locations that your scripts or commands might rely on. Have logic to handle this situation.
- The Custom Script Extension runs under the LocalSystem account.
- If you plan to use the `storageAccountName` and `storageAccountKey` properties, these properties must be collocated in `protectedSettings`.

## Extension schema

The Custom Script Extension configuration specifies things like script location and the command to be run. You can store this configuration in configuration files, specify it on the command line, or specify it in an Azure Resource Manager template.

You can store sensitive data in a protected configuration, which is encrypted and only decrypted inside the virtual machine. The protected configuration is useful when the execution command includes secrets such as a password or a shared access signature (SAS) file reference. Here's an example:

```
{  
    "apiVersion": "2018-06-01",  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "name": "virtualMachineName/config-app",  
    "location": "[resourceGroup().location]",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'),copyindex())]",  
        "[variables('musicstoresqlName')]"  
    ],  
    "tags": {  
        "displayName": "config-app"  
    },  
    "properties": {  
        "publisher": "Microsoft.Compute",  
        "type": "CustomScriptExtension",  
        "typeHandlerVersion": "1.10",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "timestamp": 123456789  
        },  
        "protectedSettings": {  
            "commandToExecute": "myExecutionCommand",  
            "storageAccountName": "myStorageAccountName",  
            "storageAccountKey": "myStorageAccountKey",  
            "managedIdentity": {},  
            "fileUris": [  
                "script location"  
            ]  
        }  
    }  
}
```

### NOTE

The `managedIdentity` property *must not* be used in conjunction with the `storageAccountName` or `storageAccountKey` property.

Only one version of an extension can be installed on a VM at a point in time. Specifying a custom script twice in the same Azure Resource Manager template for the same VM will fail.

You can use this schema inside the VM resource or as a standalone resource. The name of the resource has to be in the format `virtualMachineName/extensionName`, if this extension is used as a standalone resource in the Azure Resource Manager template.

### Property values

NAME	VALUE OR EXAMPLE	DATA TYPE
<code>apiVersion</code>	2015-06-15	date
<code>publisher</code>	Microsoft.Compute	string
<code>type</code>	CustomScriptExtension	string
<code>typeHandlerVersion</code>	1.10	int
<code>fileUris</code>	<a href="https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1">https://raw.githubusercontent.com/Microsoft/dotnet-core-sample-templates/master/dotnet-core-music-windows/scripts/configure-music-app.ps1</a>	
<code>timestamp</code>	123456789	32-bit integer
<code>commandToExecute</code>	powershell -ExecutionPolicy Unrestricted -File configure-music-app.ps1	string

NAME	VALUE OR EXAMPLE	DATA TYPE
<code>storageAccountName</code>	<code>examplestorageacct</code>	string
<code>storageAccountKey</code>	<code>TmJK/1N3AbAZ3q/+hOxoi/173zOqsaxXDhqa9t8singsUpXQp2DQIBuv2Tifp60cE/OaHsJZmQZ7teQfczQj8hg==</code>	
<code>managedIdentity</code>	<p>{ } or  { "clientId": "31b403aa-c364-4240-a7ff-d85fb6cd7232" }</p> <p>Or</p> <p>{ "objectId": "12dd289c-0583-46e5-b9b4-115d5c19ef4b" }</p>	JSON object

#### NOTE

These property names are case-sensitive. To avoid deployment problems, use the names as shown here.

#### Property value details

PROPERTY	OPTIONAL OR REQUIRED	DETAILS
<code>fileUris</code>	Optional	URLs for files to be downloaded. If URLs are sensitive (for example, they contain keys), this field should be specified in <code>protectedSettings</code> .
<code>commandToExecute</code>	Required	The entry point script to run. Use this property if your command contains secrets such as passwords or if your file URLs are sensitive.
<code>timestamp</code>	Optional	Change this value only to trigger a rerun of the script. Any integer value is acceptable, as long as it's different from the previous value.
<code>storageAccountName</code>	Optional	The name of storage account. If you specify storage credentials, all <code>fileUris</code> values must be URLs for Azure blobs.
<code>storageAccountKey</code>	Optional	The access key of the storage account.
<code>managedIdentity</code>	Optional	<p>The <a href="#">managed identity</a> for downloading files:</p> <p><code>clientId</code> (optional, string): The client ID of the managed identity.</p> <p><code>objectId</code> (optional, string): The object ID of the managed identity.</p>

You can set the following values in either public or protected settings. The extension will reject any configuration where these values are set in both public and protected settings.

- `commandToExecute`
- `fileUris`

Using public settings might be useful for debugging, but we recommend that you use protected settings.

Public settings are sent in clear text to the VM where the script will be run. Protected settings are encrypted through a key known only to Azure and the VM. The settings are saved to the VM as they were sent. That is, if the settings were encrypted, they're saved encrypted on the VM. The certificate that's used to decrypt the encrypted values is stored on the VM. The certificate is also used to decrypt settings (if necessary) at runtime.

#### Property: `managedIdentity`

#### NOTE

This property *must* be specified in protected settings only.

The Custom Script Extension (version 1.10 and later) supports [managed identities](#) for downloading files from URLs provided in the `fileUris` setting. It allows the Custom Script Extension to access Azure Storage private blobs or containers without the user having to pass secrets like SAS tokens or storage account keys.

To use this feature, the user must add a [system-assigned](#) or [user-assigned](#) identity to the VM or virtual machine scale set where the Custom Script Extension is expected to run. The user must then [grant the managed identity access to the Azure Storage container or blob](#).

To use the system-assigned identity on the target VM or virtual machine scale set, set `managedIdentity` to an empty JSON object.

Example:

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.ps1"],  
    "commandToExecute": "powershell.exe script1.ps1",  
    "managedIdentity": {}  
}
```

To use the user-assigned identity on the target VM or virtual machine scale set, configure `managedIdentity` with the client ID or the object ID of the managed identity.

Examples:

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.ps1"],  
    "commandToExecute": "powershell.exe script1.ps1",  
    "managedIdentity": { "clientId": "31b403aa-c364-4240-a7ff-d85fb6cd7232" }  
}
```

```
{  
    "fileUris": ["https://mystorage.blob.core.windows.net/privatecontainer/script1.ps1"],  
    "commandToExecute": "powershell.exe script1.ps1",  
    "managedIdentity": { "objectId": "12dd289c-0583-46e5-b9b4-115d5c19ef4b" }  
}
```

#### NOTE

The `managedIdentity` property *must not* be used in conjunction with the `storageAccountName` or `storageAccountKey` property.

## Template deployment

You can deploy Azure VM extensions by using Azure Resource Manager templates. The JSON schema detailed in the previous section can be used in an Azure Resource Manager template to run the Custom Script Extension during the template's deployment. The following samples show how to use the Custom Script Extension:

- [Tutorial: Deploy virtual machine extensions with Azure Resource Manager templates](#)
- [Deploy Two Tier Application on Windows and Azure SQL Database](#)

## PowerShell deployment

You can use the `Set-AzVMCustomScriptExtension` command to add the Custom Script Extension to an existing virtual machine. For more information, see [Set-AzVMCustomScriptExtension](#).

```
Set-AzVMCustomScriptExtension -ResourceGroupName <resourceGroupName> `  
    -VMName <vmName> `  
    -Location myLocation `  
    -FileUri <fileUrl> `  
    -Run 'myScript.ps1' `  
    -Name DemoScriptExtension
```

## Additional examples

### Using multiple scripts

In this example, you're using three scripts to build your server. The `commandToExecute` property calls the first script. You then have options on how the others are called. For example, you can have a master script that controls the execution, with the right error handling, logging, and state management. The scripts are downloaded to the local machine for execution.

For example, in `1_Add_Tools.ps1`, you would call `2_Add_Features.ps1` by adding `.\2_Add_Features.ps1` to the script. You would repeat this process for the other scripts that you define in `$settings`.

```

$fileUri = @("https://xxxxxxxx.blob.core.windows.net/buildServer1/1_Add_Tools.ps1",
"https://xxxxxxxx.blob.core.windows.net/buildServer1/2_Add_Features.ps1",
"https://xxxxxxxx.blob.core.windows.net/buildServer1/3_CompleteInstall.ps1")

$settings = @{$fileUris" = $fileUri};

$storageAcctName = "xxxxxxxx"
$storageKey = "1234ABCD"
$protectedSettings = @{$storageAcctName" = $storageAcctName; "storageAccountKey" = $storageKey;
"commandToExecute" = "powershell -ExecutionPolicy Unrestricted -File 1_Add_Tools.ps1"};

#run command
Set-AzVMExtension -ResourceGroupName <resourceGroupName> ` 
-Location <locationName> ` 
-VMName <vmName> ` 
-Name "buildserver1" ` 
-Publisher "Microsoft.Compute" ` 
-ExtensionType "CustomScriptExtension" ` 
-TypeHandlerVersion "1.10" ` 
-Settings $settings ` 
-ProtectedSettings $protectedSettings

```

### Running scripts from a local share

In this example, you might want to use a local Server Message Block (SMB) server for your script location. You then don't need to provide any other settings, except `commandToExecute`.

```

$protectedSettings = @{$"commandToExecute" = "powershell -ExecutionPolicy Unrestricted -File
\\filesrv\build\serverUpdate1.ps1"};

Set-AzVMExtension -ResourceGroupName <resourceGroupName> ` 
-Location <locationName> ` 
-VMName <vmName> ` 
-Name "serverUpdate" ` 
-Publisher "Microsoft.Compute" ` 
-ExtensionType "CustomScriptExtension" ` 
-TypeHandlerVersion "1.10" ` 
-ProtectedSettings $protectedSettings

```

### Running a custom script more than once by using the CLI

The Custom Script Extension handler will prevent rerunning a script if the *exact* same settings have been passed. This behavior prevents accidental rerunning, which might cause unexpected behaviors if the script isn't idempotent. You can confirm if the handler has blocked the rerunning by looking at `C:\Windows\Azure\Logs\Plugins\Microsoft.Compute.CustomScriptExtension<HandlerVersion>\CustomScriptHandler.log` and searching for a warning like this one:

```
Current sequence number, <SequenceNumber>, is not greater than the sequence number of the most recently
executed configuration. Exiting...
```

If you want to run the Custom Script Extension more than once, you can do that only under these conditions:

- The extension's `Name` parameter is the same as the previous deployment of the extension.
- You've updated the configuration. You can add a dynamic property to the command, such as a timestamp. If the handler detects a change in the configuration settings, it will consider that change as an explicit desire to rerun the script.

Alternatively, you can set the `ForceUpdateTag` property to `true`.

### Using Invoke-WebRequest

If you're using `Invoke-WebRequest` in your script, you must specify the parameter `-UseBasicParsing`. If you don't specify the parameter, you'll get the following error when checking the detailed status:

```
The response content cannot be parsed because the Internet Explorer engine is not available, or Internet
Explorer's first-launch configuration is not complete. Specify the UseBasicParsing parameter and try again.
```

## Virtual machine scale sets

If you deploy the Custom Script Extension from the Azure portal, you don't have control over the expiration of the SAS token for accessing the script in your storage account. The result is that the initial deployment works, but when the storage account's SAS token expires, any subsequent scaling operation fails because the Custom Script Extension can no longer access the storage account.

We recommend that you use [PowerShell](#), the [Azure CLI](#), or an Azure Resource Manager template when you deploy the Custom Script Extension on a virtual machine scale set. This way, you can choose to use a managed identity or have direct control of the expiration of the SAS token for accessing the script in your storage account for as long as you need.

## Troubleshoot and support

You can retrieve data about the state of extension deployments from the Azure portal and by using the Azure PowerShell module. To see the deployment state of extensions for a VM, run the following command:

```
Get-AzVMExtension -ResourceGroupName <resourceGroupName> -VMName <vmName> -Name myExtensionName
```

Extension output is logged to files found under the following folder on the target virtual machine:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.Compute.CustomScriptExtension
```

The specified files are downloaded into the following folder on the target virtual machine:

```
C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.*\Downloads\<n>
```

In the preceding path, `<n>` is a decimal integer that might change between executions of the extension. The `1.*` value matches the actual, current `typeHandlerVersion` value of the extension. For example, the actual directory could be `C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.8\Downloads\2`.

When you run the `commandToExecute` command, the extension sets this directory (for example, `...\Downloads\2`) as the current working directory. This process enables the use of relative paths to locate the files downloaded via the `fileURIs` property. Here are examples of downloaded files:

URI IN FILEURIS	RELATIVE DOWNLOAD LOCATION	ABSOLUTE DOWNLOAD LOCATION <sup>1</sup>
<code>https://someAcct.blob.core.windows.net/.scripts/myscript.ps1 cript.ps1</code>		<code>C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.8\Downloads\2\myscript.ps1 cript.ps1</code>
<code>https://someAcct.blob.core.windows.net/.topLevel.ps1 Level.ps1</code>		<code>C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.8\Downloads\2\topLevel.ps1 Level.ps1</code>

<sup>1</sup> The absolute directory paths change over the lifetime of the VM, but not within a single execution of the Custom Script Extension.

Because the absolute download path might vary over time, it's better to opt for relative script/file paths in the `commandToExecute` string, whenever possible. For example:

```
"commandToExecute": "powershell.exe . . . -File \"./scripts/myscript.ps1\""
```

Path information after the first URI segment is kept for files downloaded via the `fileURIs` property list. As shown in the earlier table, downloaded files are mapped into download subdirectories to reflect the structure of the `fileURIs` values.

## Support

If you need help with any part of this article, you can contact the Azure experts at [Azure Community Support](#).

You can also file an Azure support incident. Go to the [Azure support site](#) and select **Get support**. For information about using Azure support, read the [Microsoft Azure support FAQ](#).

# Microsoft Antimalware Extension for Windows

9/21/2022 • 5 minutes to read • [Edit Online](#)

## Overview

The modern threat landscape for cloud environments is extremely dynamic, increasing the pressure on business IT cloud subscribers to maintain effective protection in order to meet compliance and security requirements. Microsoft Antimalware for Azure is free real-time protection capability that helps identify and remove viruses, spyware, and other malicious software, with configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems. The solution is built on the same antimalware platform as Microsoft Security Essentials (MSE), Microsoft Forefront Endpoint Protection, Microsoft System Center Endpoint Protection, Windows Intune, and Windows Defender for Windows 8.0 and higher. Microsoft Antimalware for Azure is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. You can deploy protection based on the needs of your application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring.

## Prerequisites

### Operating system

The Microsoft Antimalware for Azure solution includes the Microsoft Antimalware Client, and Service, Antimalware classic deployment model, Antimalware PowerShell cmdlets, and Azure Diagnostics Extension. The Microsoft Antimalware solution is supported on Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2 operating system families. It is not supported on the Windows Server 2008 operating system, and also is not supported in Linux.

Windows Defender is the built-in Antimalware enabled in Windows Server 2016. The Windows Defender Interface is also enabled by default on some Windows Server 2016 SKU's. The Azure VM Antimalware extension can still be added to a Windows Server 2016 Azure VM with Windows Defender, but in this scenario the extension will apply any optional configuration policies to be used by Windows Defender, the extension will not deploy any additional antimalware service. For more information, see [Update to Azure Antimalware Extension for Cloud Services](#).

### Internet connectivity

The Microsoft Antimalware for Windows requires that the target virtual machine is connected to the internet to receive regular engine and signature updates.

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment configuration such as onboarding to Azure Antimalware.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the VM extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "[concat(parameters('vmName'), '/', parameters('vmExtensionName'))]",
  "apiVersion": "2019-07-01",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', parameters('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Security",
    "type": "IaaSAntimalware",
    "typeHandlerVersion": "1.3",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "AntimalwareEnabled": "true",
      "Exclusions": {
        "Extensions": ".ext1;.ext2",
        "Paths": "c:\excluded-path-1;c:\excluded-path-2",
        "Processes": "excludedproc1.exe;excludedproc2.exe"
      },
      "RealtimeProtectionEnabled": "true",
      "ScheduledScanSettings": {
        "isEnabled": "true",
        "scanType": "Quick",
        "day": "7",
        "time": "120"
      }
    },
    "protectedSettings": null
  }
}
```

You must include, at a minimum, the following content to enable the Microsoft Antimalware extension:

```
{ "AntimalwareEnabled": true }
```

Microsoft Antimalware JSON configuration sample:

```
{
  "AntimalwareEnabled": true,
  "RealtimeProtectionEnabled": true,
  "ScheduledScanSettings": {
    "isEnabled": true,
    "day": 1,
    "time": 120,
    "scanType": "Full"
  },
  "Exclusions": {
    "Extensions": ".ext1;.ext2",
    "Paths": "c:\excluded-path-1;c:\excluded-path-2",
    "Processes": "excludedproc1.exe;excludedproc2.exe"
  }
}
```

#### AntimalwareEnabled

- required parameter
- Values: true/false
  - true = Enable
  - false = Error out, as false is not a supported value

#### RealtimeProtectionEnabled

- Values: true/false, default is true
  - true = Enable
  - false = Disable

## ScheduledScanSettings

- isEnabled = true/false
- day = 0-8 (0=daily, 1=Sunday, 2=Monday, ..., 7=Saturday, 8=Disabled)
- time = 0-1440 (measured in minutes after midnight - 60->1AM, 120 -> 2AM, ... )
- scanType = Quick/Full, default is Quick
- If isEnabled = true is the only setting provided, the following defaults are set: day=7 (Saturday), time=120 (2 AM), scanType="Quick"

## Exclusions

- Multiple exclusions in the same list are specified by using semicolon delimiters
- If no exclusions are specified, then the existing exclusions, if any, are overwritten by blank on the system

## PowerShell deployment

Depends on your type of deployment, use the corresponding commands to deploy the Azure Antimalware virtual machine extension to an existing virtual machine.

- [Azure Resource Manager based Virtual Machine](#)
- [Azure Service Fabric Clusters](#)
- [Classic Cloud Service](#)

## Troubleshoot and support

### Troubleshoot

Microsoft Antimalware extension logs are available at -

%Systemdrive%\WindowsAzure\Logs\Plugins\Microsoft.Azure.Security.IaaSAntimalware(Or  
PaaSAntimalware)\1.5.5.x(version#)\CommandExecution.log

### Error codes and their meanings

ERROR CODE	MEANING	POSSIBLE ACTION
-2147156224	MSI is busy with different installation	Try running installation later
-2147156221	MSE setup already running	Run only one instance at a time
-2147156208	Low disk space < 200 MB	Delete unused files, and retry installation
-2147156187	Last installation, upgrade, update, or uninstall requested reboot	Reboot, and retry installation
-2147156121	Setup tried to remove competitor product. But competitor product uninstall failed	Try to remove the competitor product manually, reboot, and retry installation
-2147156116	Policy file validation failed	Make sure you pass a valid policy XML file to setup
-2147156095	Setup couldn't start the Antimalware service	Verify all binaries are correctly signed, and right licensing file is installed

ERROR CODE	MEANING	POSSIBLE ACTION
-2147023293	A fatal error occurred during installation. In most cases, it will. Epp.msi, can't register\start\stop AM service or mini filter driver	MSI logs from EPP.msi are required here for future investigation
-2147023277	Installation package could not be opened	Verify that the package exists, and is accessible, or contact the application vendor to verify that this is a valid Windows Installer package
-2147156109	Windows Defender is required as a prerequisite	
-2147205073	The websso issuer is not supported	
-2147024893	The system cannot find the path specified	
-2146885619	Not a cryptographic message or the cryptographic message is not formatted correctly	
-1073741819	The instruction at 0x%p referenced memory at 0x%p. The memory could not be %s	
1	Incorrect Function	

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#), and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# VM Snapshot Linux extension for Azure Backup

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure Backup provides support for backing up workloads from on-premises to cloud and backing up cloud resources to Recovery Services vault. Azure Backup uses VM snapshot extension to take an application consistent backup of the Azure virtual machine without the need to shutdown the VM. VM Snapshot Linux extension is published and supported by Microsoft as part of Azure Backup service. Azure Backup will install the extension as part of first scheduled backup triggered post enabling backup. This document details the supported platforms, configurations, and deployment options for the VM Snapshot extension.

The VMSnapshot extension appears in the Azure portal only for non-managed VMs.

## Prerequisites

### Operating system

For a list of supported operating systems, please refer to [Operating Systems supported by Azure Backup](#)

## Extension schema

The following JSON shows the schema for the VM snapshot extension. The extension requires the task ID - this identifies the backup job which triggered snapshot on the VM, status blob uri - where status of the snapshot operation is written, scheduled start time of the snapshot, logs blob uri - where logs corresponding to snapshot task are written, objstr- representation of VM disks and meta data. Because these settings should be treated as sensitive data, it should be stored in a protected setting configuration. Azure VM extension protected setting data is encrypted, and only decrypted on the target virtual machine. Please note that these settings are recommended to be passed from Azure Backup service only as part of Backup job.

```
{
  "type": "extensions",
  "name": "VMSnapshot",
  "location": "<myLocation>",
  "properties": {
    "publisher": "Microsoft.RecoveryServices",
    "type": "VMSnapshot",
    "typeHandlerVersion": "1.9",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "locale": "<location>",
      "taskId": "<taskId used by Azure Backup service to communicate with extension>",
      "commandToExecute": "snapshot",
      "commandStartTimeUTCTicks": "<scheduled start time of the snapshot task>",
      "vmType": "microsoft.compute/virtualmachines"
    },
    "protectedSettings": {
      "objectStr": "<blob SAS uri representation of VM sent by Azure Backup service to extension>",
      "logsBlobUri": "<blob uri where logs of command execution by extension are written to>",
      "statusBlobUri": "<blob uri where status of the command executed by extension is written>"
    }
  }
}
```

## Property values

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2015-06-15	date
taskId	e07354cf-041e-4370-929f-25a319ce8933_1	string
commandStartTimeUTCTicks	6.36458E+17	string
locale	en-us	string
objectStr	Encoding of sas uri array- "blobSASUri": ["https://sopattna5365.blob.core.windows.net/vhds/vmubuntu1404ltsc20165 2903941.vhd?sv=2014-02- 14&sr=b&sig=TywkROXL1zvhXcLujtC ut8g3jTpgbE6JpSWRLZxAdtA%3D&st= 2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna8461.blob.core.windows.net/vhds/vmubuntu1404ltsc- 20160629-122418.vhd?sv=2014-02- 14&sr=b&sig=5S0A6YDWvVwqPAkz WXVy%2BS%2FqMwzFMbamT5upwx0 5v8Q%3D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna8461.blob.core.windows.net/bootdiagnostics-vmubuntu1- deb58392-ed5e-48be-9228- ff681b0cd3ee/vmubuntu1404ltsc- 20160629-122541.vhd?sv=2014-02- 14&sr=b&sig=X0Me2djByksBBMVXM GIUrcycvhQSfjYvqKLeRA7nBD4%3D&s t=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna5365.blob.core.windows.net/vhds/vmubuntu1404ltsc- 20160701-163922.vhd?sv=2014-02- 14&sr=b&sig=oXvtK2IXCNqWv7fpjc7 TAzFDpc1GoXtT7r%2BC%2BNIaork%3 D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna5365.blob.core.windows.net/vhds/vmubuntu1404ltsc- 20170705-124311.vhd?sv=2014-02- 14&sr=b&sig=ZUM9d28Mvvm%2Ffrh J71TFZh0Ni90m38bBs3zMI%2FQ9rs0 %3D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw"]	string

Name	Value / Example	Data Type
logsBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DbwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DbwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string
statusBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5IduB%2FILktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5IduB%2FILktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. However, the recommended way of adding a VM snapshot extension to a virtual machine is by enabling backup on the virtual machine. This can be achieved through a Resource Manager template. A sample Resource Manager template that enables backup on a virtual machine can be found on the [Azure Quick Start Gallery](#).

## Azure CLI deployment

The Azure CLI can be used to enable backup on a virtual machine. Post enable backup, first scheduled backup job will install the Vm snapshot extension on the VM.

```
az backup protection enable-for-vm \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--vm myVM \
--policy-name DefaultPolicy
```

## Azure PowerShell deployment

Azure PowerShell can be used to enable backup on a virtual machine. Once the backup is configured, first scheduled backup job will install the Vm snapshot extension on the VM.

```
$targetVault = Get-AzRecoveryServicesVault -ResourceGroupName "myResourceGroup" -Name "myRecoveryServicesVault"
$pol = Get-AzRecoveryServicesBackupProtectionPolicy Name DefaultPolicy -VaultId $targetVault.ID
Enable-AzRecoveryServicesBackupProtection -Policy $pol -Name "myVM" -ResourceGroupName "myVMResourceGroup" -VaultId $targetVault.ID
```

# Troubleshoot and support

## Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file:

```
/var/log/waagent.log
```

## Error codes and their meanings

Troubleshooting information can be found on the [Azure VM backup troubleshooting guide](#).

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# VM Snapshot Windows extension for Azure Backup

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure Backup provides support for backing up workloads from on-premises to cloud and backing up cloud resources to Recovery Services vault. Azure Backup uses VM snapshot extension to take an application consistent backup of the Azure virtual machine without the need to shutdown the VM. VM Snapshot extension is published and supported by Microsoft as part of Azure Backup service. Azure Backup will install the extension as part of first scheduled backup triggered post enabling backup. This document details the supported platforms, configurations, and deployment options for the VM Snapshot extension.

The VMSnapshot extension appears in the Azure portal only for non-managed VMs.

## Prerequisites

### Operating system

For a list of supported operating systems, refer to [Operating Systems supported by Azure Backup](#)

## Extension schema

The following JSON shows the schema for the VM snapshot extension. The extension requires the task ID - this identifies the backup job which triggered snapshot on the VM, status blob uri - where status of the snapshot operation is written, scheduled start time of the snapshot, logs blob uri - where logs corresponding to snapshot task are written, objstr- representation of VM disks and meta data. Because these settings should be treated as sensitive data, it should be stored in a protected setting configuration. Azure VM extension protected setting data is encrypted, and only decrypted on the target virtual machine. Note that these settings are recommended to be passed from Azure Backup service only as part of Backup job.

```
{
  "type": "extensions",
  "name": "VMSnapshot",
  "location": "<myLocation>",
  "properties": {
    "publisher": "Microsoft.Azure.RecoveryServices",
    "type": "VMSnapshot",
    "typeHandlerVersion": "1.9",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "locale": "<location>",
      "taskId": "<taskId used by Azure Backup service to communicate with extension>",
      "commandToExecute": "snapshot",
      "commandStartTimeUTCTicks": "<scheduled start time of the snapshot task>",
      "vmType": "microsoft.compute/virtualmachines"
    },
    "protectedSettings": {
      "objectStr": "<blob SAS uri representation of VM sent by Azure Backup service to extension>",
      "logsBlobUri": "<blob uri where logs of command execution by extension are written to>",
      "statusBlobUri": "<blob uri where status of the command executed by extension is written>"
    }
  }
}
```

## Property values

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2015-06-15	date
taskId	e07354cf-041e-4370-929f-25a319ce8933_1	string
commandStartTimeUTCTicks	6.36458E+17	string
locale	en-us	string
objectStr	Encoding of sas uri array- "blobSASUri": ["https://sopattna5365.blob.core.windows.net/vhds/vmwin1404ltsc20165290 3941.vhd?sv=2014-02- 14&sr=b&sig=TywkROXL1zvhXcLujtC ut8g3jTpgbE6JpSWRLZxAdtA%3D&st= 2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna8461.blob.core.windows.net/vhds/vmwin1404ltsc- 20160629-122418.vhd?sv=2014-02- 14&sr=b&sig=5S0A6YDWvVwqPAkz WXVy%2BS%2FqMwzFMbamT5upwx0 5v8Q%3D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna8461.blob.core.windows.net/bootdiagnostics-vmwintu1- deb58392-ed5e-48be-9228- ff681b0cd3ee/vmubuntu1404ltsc- 20160629-122541.vhd?sv=2014-02- 14&sr=b&sig=X0Me2djByksBBMVXM GIUrcycvhQSfjYvqKLeRA7nBD4%3D&s t=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna5365.blob.core.windows.net/vhds/vmwin1404ltsc- 20160701-163922.vhd?sv=2014-02- 14&sr=b&sig=oXvtK2IXCNqWv7fpjc7 TAzFDpc1GoXtT7r%2BC%2BNIaork%3 D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw", "https://sopattna5365.blob.core.windows.net/vhds/vmwin1404ltsc- 20170705-124311.vhd?sv=2014-02- 14&sr=b&sig=ZUM9d28Mvvm%2Ffrh J71TFZh0Ni90m38bBs3zMI%2FQ9rs0 %3D&st=2017-11- 09T14%3A23%3A28Z&se=2017-11- 09T17%3A38%3A28Z&sp=rw"]	string

Name	Value / Example	Data Type
logsBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DbwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Logs.txt?sv=2014-02-14&amp;sr=b&amp;sig=DbwYhwfeAC5YJzISgxoKk%2FEWQq2AO1vS1E0rDW%2FlsBw%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string
statusBlobUri	<a href="https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5IduB%2FILktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw">https://seapod01coord1exsapk732.blob.core.windows.net/bcdextensionlogs-d45d8a1c-281e-4bc8-9d30-3b25176f68ea/sopattnavmubuntu1404ltsc.v2.Status.txt?sv=2014-02-14&amp;sr=b&amp;sig=96RZBpTKCjmV7QFeXm5IduB%2FILktwGbLwbWg6lh96Ao%3D&amp;st=2017-11-09T14%3A33%3A29Z&amp;se=2017-11-09T17%3A38%3A29Z&amp;sp=rw</a>	string

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. However, the recommended way of adding a VM snapshot extension to a virtual machine is by enabling backup on the virtual machine. This can be achieved through a Resource Manager template. A sample Resource Manager template that enables backup on a virtual machine can be found on the [Azure Quick Start Gallery](#).

## Azure CLI deployment

The Azure CLI can be used to enable backup on a virtual machine. Post enable backup, first scheduled backup job will install the Vm snapshot extension on the VM.

```
az backup protection enable-for-vm \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--vm myVM \
--policy-name DefaultPolicy
```

## Azure PowerShell deployment

Azure PowerShell can be used to enable backup on a virtual machine. Once the backup is configured, first scheduled backup job will install the Vm snapshot extension on the VM.

```
$targetVault = Get-AzRecoveryServicesVault -ResourceGroupName "myResourceGroup" -Name "myRecoveryServicesVault"
$pol = Get-AzRecoveryServicesBackupProtectionPolicy Name DefaultPolicy -VaultId $targetVault.ID
Enable-AzRecoveryServicesBackupProtection -Policy $pol -Name "myVM" -ResourceGroupName "myVMResourceGroup" -VaultId $targetVault.ID
```

# Troubleshoot and support

## Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file:

```
C:\Packages\Plugins\Microsoft.Azure.RecoveryServices.VMSnapshot
```

## Error codes and their meanings

Troubleshooting information can be found on the [Azure VM backup troubleshooting guide](#).

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Network Watcher Agent virtual machine extension for Linux

9/21/2022 • 2 minutes to read • [Edit Online](#)

## Overview

[Azure Network Watcher](#) is a network performance monitoring, diagnostic, and analytics service that allows monitoring for Azure networks. The Network Watcher Agent virtual machine (VM) extension is a requirement for some of the Network Watcher features on Azure VMs, such as capturing network traffic on demand, and other advanced functionality.

This article details the supported platforms and deployment options for the Network Watcher Agent VM extension for Linux. Installation of the agent does not disrupt, or require a reboot, of the VM. You can deploy the extension into virtual machines that you deploy. If the virtual machine is deployed by an Azure service, check the documentation for the service to determine whether or not it permits installing extensions in the virtual machine.

## Prerequisites

### Operating system

The Network Watcher Agent extension can be configured for the following Linux distributions:

DISTRIBUTION	VERSION
Ubuntu	12+
Debian	7 and 8
Red Hat	6 and 7
Oracle Linux	6.8+, 7 and 8+
SUSE Linux Enterprise Server	11, 12 and 15
OpenSUSE Leap	42.3+
CentOS	6.5+ and 7
CoreOS	899.17.0+

### Internet connectivity

Some of the Network Watcher Agent functionality requires that a VM is connected to the Internet. Without the ability to establish outgoing connections, some of the Network Watcher Agent features may malfunction, or become unavailable. For more information about Network Watcher functionality that requires the agent, see the [Network Watcher documentation](#).

## Extension schema

The following JSON shows the schema for the Network Watcher Agent extension. The extension doesn't require,

or support, any user-supplied settings. The extension relies on its default configuration.

```
{  
  "type": "extensions",  
  "name": "Microsoft.Azure.NetworkWatcher",  
  "apiVersion": "[variables('apiVersion')]",  
  "location": "[resourceGroup().location]",  
  "dependsOn": [  
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"  
  ],  
  "properties": {  
    "publisher": "Microsoft.Azure.NetworkWatcher",  
    "type": "NetworkWatcherAgentLinux",  
    "typeHandlerVersion": "1.4",  
    "autoUpgradeMinorVersion": true  
  }  
}
```

## Property values

NAME	VALUE / EXAMPLE
apiVersion	2015-06-15
publisher	Microsoft.Azure.NetworkWatcher
type	NetworkWatcherAgentLinux
typeHandlerVersion	1.4

## Template deployment

You can deploy Azure VM extensions with an Azure Resource Manager template. To deploy the Network Watcher Agent extension, use the previous json schema in your template.

## Azure classic CLI deployment

### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

The following example deploys the Network Watcher Agent VM extension to an existing VM deployed through the classic deployment model:

```
azure config mode asm  
azure vm extension set myVM1 NetworkWatcherAgentLinux Microsoft.Azure.NetworkWatcher 1.4
```

## Azure CLI deployment

The following example deploys the Network Watcher Agent VM extension to an existing VM deployed through Resource Manager:

```
az vm extension set --resource-group myResourceGroup1 --vm-name myVM1 --name NetworkWatcherAgentLinux --publisher Microsoft.Azure.NetworkWatcher --version 1.4
```

## Troubleshooting and support

### Troubleshooting

You can retrieve data about the state of extension deployments using either the Azure portal or Azure CLI.

The following example shows the deployment state of the NetworkWatcherAgentLinux extension for a VM deployed through Resource Manager, using the Azure CLI:

```
az vm extension show --name NetworkWatcherAgentLinux --resource-group myResourceGroup1 --vm-name myVM1
```

### Support

If you need more help at any point in this article, you can refer to the [Network Watcher documentation](#), or contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select **Get support**. For information about using Azure Support, see the [Microsoft Azure support FAQ](#).

# Network Watcher Agent virtual machine extension for Windows

9/21/2022 • 2 minutes to read • [Edit Online](#)

## Overview

[Azure Network Watcher](#) is a network performance monitoring, diagnostic, and analytics service that allows monitoring of Azure networks. The Network Watcher Agent virtual machine extension is a requirement for capturing network traffic on demand, and other advanced functionality on Azure virtual machines.

This document details the supported platforms and deployment options for the Network Watcher Agent virtual machine extension for Windows. Installation of the agent does not disrupt, or require a reboot, of the virtual machine. You can deploy the extension into virtual machines that you deploy. If the virtual machine is deployed by an Azure service, check the documentation for the service to determine whether or not it permits installing extensions in the virtual machine.

## Prerequisites

### Operating system

The Network Watcher Agent extension for Windows can be run against Windows Server 2008 R2, 2012, 2012 R2, 2016 and 2019 releases. Nano Server is not supported at this time.

### Internet connectivity

Some of the Network Watcher Agent functionality requires that the target virtual machine be connected to the Internet. Without the ability to establish outgoing connections, the Network Watcher Agent will not be able to upload packet captures to your storage account. For more details, please see the [Network Watcher documentation](#).

## Extension schema

The following JSON shows the schema for the Network Watcher Agent extension. The extension neither requires, nor supports, any user-supplied settings, and relies on its default configuration.

```
{
  "type": "extensions",
  "name": "Microsoft.Azure.NetworkWatcher",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.NetworkWatcher",
    "type": "NetworkWatcherAgentWindows",
    "typeHandlerVersion": "1.4",
    "autoUpgradeMinorVersion": true
  }
}
```

## Property values

NAME	VALUE / EXAMPLE
apiVersion	2015-06-15
publisher	Microsoft.Azure.NetworkWatcher
type	NetworkWatcherAgentWindows
typeHandlerVersion	1.4

## Template deployment

You can deploy Azure VM extensions with Azure Resource Manager templates. You can use the JSON schema detailed in the previous section in an Azure Resource Manager template to run the Network Watcher Agent extension during an Azure Resource Manager template deployment.

## PowerShell deployment

Use the `Set-AzVMExtension` command to deploy the Network Watcher Agent virtual machine extension to an existing virtual machine:

```
Set-AzVMExtension ` 
-ResourceGroupName "myResourceGroup1" ` 
-Location "WestUS" ` 
-VMName "myVM1" ` 
-Name "networkWatcherAgent" ` 
-Publisher "Microsoft.Azure.NetworkWatcher" ` 
-Type "NetworkWatcherAgentWindows" ` 
-TypeHandlerVersion "1.4"
```

## Troubleshooting and support

### Troubleshooting

You can retrieve data about the state of extension deployments from the Azure portal and PowerShell. To see the deployment state of extensions for a given VM, run the following command using the Azure PowerShell module:

```
Get-AzVMExtension -ResourceGroupName myResourceGroup1 -VMName myVM1 -Name networkWatcherAgent
```

Extension execution output is logged to files found in the following directory:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.Azure.NetworkWatcher.NetworkWatcherAgentWindows\
```

### Support

If you need more help at any point in this article, you can refer to the Network Watcher User Guide documentation or contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Update the Network Watcher extension to the latest version

9/21/2022 • 4 minutes to read • [Edit Online](#)

## Overview

Azure Network Watcher is a network performance monitoring, diagnostic, and analytics service that monitors Azure networks. The Network Watcher Agent virtual machine (VM) extension is a requirement for capturing network traffic on demand and using other advanced functionality on Azure VMs. The Network Watcher extension is used by features like Connection Monitor, Connection Monitor (preview), connection troubleshoot, and packet capture.

## Prerequisites

This article assumes you have the Network Watcher extension installed in your VM.

## Latest version

The latest version of the Network Watcher extension is currently [1.4.2146.1](#).

## Update your extension using a PowerShell script

Customers with large deployments who need to update multiple VMs at once. For updating select VMs manually, please see the next section

```
<#
 .SYNOPSIS
 This script will scan all VMs in the provided subscription and upgrade any out of date
 AzureNetworkWatcherExtensions

 .DESCRIPTION
 This script should be no-op if AzureNetworkWatcherExtensions are up to date
 Requires Azure PowerShell 4.2 or higher to be installed (e.g. Install-Module AzureRM).

 .EXAMPLE
 .\UpdateVMAgentsInSub.ps1 -SubID F4BC4873-5DAB-491E-B713-1358EF4992F2 -NoUpdate

 #>
 [CmdletBinding()]
 param(
     [Parameter(Mandatory=$true)]
     [string] $SubID,
     [Parameter(Mandatory=$false)]
     [Switch] $NoUpdate = $false,
     [Parameter(Mandatory=$false)]
     [string] $MinVersion = "1.4.1974.1"
 )

 function NeedsUpdate($version)
 {
     if ($version -eq $MinVersion)
     {
         return $false
     }
 }
```

```

$lessThan = $true;
$versionParts = $version -split '\.';
$minVersionParts = $MinVersion -split '\.';
for ($i = 0; $i -lt $versionParts.Length; $i++)
{
    if ([int]$versionParts[$i] -gt [int]$minVersionParts[$i])
    {
        $lessThan = $false;
        break;
    }
}

return $lessThan
}

Write-Host "Scanning all VMs in the subscription: $($SubID)"
Select-AzSubscription -SubscriptionId $SubID;
$vms = Get-AzVM;
$foundVMs = $false;
Write-Host "Starting VM search, this may take a while"

foreach ($vmName in $vms)
{
    # Get Detailed VM info
    $vm = Get-AzVM -ResourceGroupName $vmName.ResourceGroupName -Name $vmName.name -Status;
    $isWindows = $vm.OsVersion -match "Windows";
    foreach ($extension in $vm.Extensions)
    {
        if ($extension.Name -eq "AzureNetworkWatcherExtension")
        {
            if (NeedsUpdate($extension.TypeHandlerVersion))
            {
                $foundVMs = $true;
                if (-not ($NoUpdate))
                {
                    Write-Host "Found VM that needs to be updated:
subscriptions/$($SubID)/resourceGroups/$($vm.ResourceGroupName)/providers/Microsoft.Compute/virtualMachines/
 $($vm.Name) -> Updating " -NoNewline
                    Remove-AzVMEExtension -ResourceGroupName $vm.ResourceGroupName -VMName $vm.Name -Name
"AzureNetworkWatcherExtension" -Force
                    Write-Host "... " -NoNewline
                    $type = if ($isWindows) { "NetworkWatcherAgentWindows" } else {
"NetworkWatcherAgentLinux" };
                    Set-AzVMEExtension -ResourceGroupName $vm.ResourceGroupName -Location $vmName.Location -
VMName $vm.Name -Name "AzureNetworkWatcherExtension" -Publisher "Microsoft.Azure.NetworkWatcher" -Type $type
-typeHandlerVersion "1.4"
                    Write-Host "Done"
                }
                else
                {
                    Write-Host "Found $($if ($isWindows) {"Windows"} else {"Linux"}) VM that needs to be
updated:
subscriptions/$($SubID)/resourceGroups/$($vm.ResourceGroupName)/providers/Microsoft.Compute/virtualMachines/
 $($vm.Name)"
                }
            }
        }
    }
}

if ($foundVMs)
{
    Write-Host "Finished $($if ($NoUpdate) {"searching"} else {"updating"}) out of date
AzureNetworkWatcherExtension on VMs"
}
else
{
    Write-Host "All AzureNetworkWatcherExtensions up to date"
}

```

# Update your extension manually

To update your extension, you need to know your extension version.

## Check your extension version

You can check your extension version by using the Azure portal, the Azure CLI, or PowerShell.

### Use the Azure portal

1. Go to the Extensions pane of your VM in the Azure portal.
2. Select the **AzureNetworkWatcher** extension to see the details pane.
3. Locate the version number in the **Version** field.

### Use the Azure CLI

Run the following command from an Azure CLI prompt:

```
az vm get-instance-view --resource-group "SampleRG" --name "Sample-VM"
```

Locate "AzureNetworkWatcherExtension" in the output and identify the version number from the "TypeHandlerVersion" field in the output. Please note: Information about the extension appears multiple times in the JSON output. Please look under the "extensions" block and you should see the full version number of the extension.

You should see something like the below:

```
{  
  "status": {  
    "code": "ProvisioningState/succeeded",  
    "displayStatus": "Ready",  
    "level": "Info",  
    "message": null,  
    "time": null  
  },  
  "type": "Microsoft.Azure.NetworkWatcher.NetworkWatcherAgentWindows",  
  "typeHandlerVersion": "1.4.1654.1"  
},
```

### Use PowerShell

Run the following commands from a PowerShell prompt:

```
Get-AzVM -ResourceGroupName "SampleRG" -Name "Sample-VM" -Status
```

Locate the Azure Network Watcher extension in the output and identify the version number from the "TypeHandlerVersion" field in the output.

You should see something like the below:

```
ExtensionHandlers[1] :  
  Type : Microsoft.Azure.NetworkWatcher.NetworkWatcherAgentWindows  
  TypeHandlerVersion : 1.4.1654.1  
  Status :  
  Code : ProvisioningState/succeeded  
  Level : Info
```

## Update your extension

If your version is below the latest version mentioned above, update your extension by using any of the following options.

#### Option 1: Use PowerShell

Run the following commands:

```
#Linux command
Set-AzVMExtension -ResourceGroupName "myResourceGroup1" -Location "WestUS" -VMName "myVM1" -Name
"AzureNetworkWatcherExtension" -Publisher "Microsoft.Azure.NetworkWatcher" -Type "NetworkWatcherAgentLinux"

#Windows command
Set-AzVMExtension -ResourceGroupName "myResourceGroup1" -Location "WestUS" -VMName "myVM1" -Name "
AzureNetworkWatcherExtension" -Publisher "Microsoft.Azure.NetworkWatcher" -Type "NetworkWatcherAgentWindows"
-ForceRerun "True"
```

If that doesn't work, remove and install the extension again, using the steps below. This will automatically add the latest version.

Removing the extension

```
#Same command for Linux and Windows
Remove-AzVMExtension -ResourceGroupName "SampleRG" -VMName "Sample-VM" -Name "AzureNetworkWatcherExtension"
```

Installing the extension again

```
#Linux command
Set-AzVMExtension -ResourceGroupName "SampleRG" -Location "centralus" -VMName "Sample-VM" -Name
"AzureNetworkWatcherExtension" -Publisher "Microsoft.Azure.NetworkWatcher" -Type "NetworkWatcherAgentLinux"
-typeHandlerVersion "1.4"

#Windows command
Set-AzVMExtension -ResourceGroupName "SampleRG" -Location "centralus" -VMName "Sample-VM" -Name
"AzureNetworkWatcherExtension" -Publisher "Microsoft.Azure.NetworkWatcher" -Type
"NetworkWatcherAgentWindows" -typeHandlerVersion "1.4"
```

#### Option 2: Use the Azure CLI

Force an upgrade.

```
#Linux command
az vm extension set --resource-group "myResourceGroup1" --vm-name "myVM1" --name "NetworkWatcherAgentLinux"
--publisher "Microsoft.Azure.NetworkWatcher" --force-update

#Windows command
az vm extension set --resource-group "myResourceGroup1" --vm-name "myVM1" --name
"NetworkWatcherAgentWindows" --publisher "Microsoft.Azure.NetworkWatcher" --force-update
```

If that doesn't work, remove and install the extension again, and follow these steps to automatically add the latest version.

Remove the extension.

```
#Same for Linux and Windows
az vm extension delete --resource-group "myResourceGroup1" --vm-name "myVM1" -n
"AzureNetworkWatcherExtension"
```

Install the extension again.

```
#Linux command  
az vm extension set --resource-group "DALANDEMO" --vm-name "Linux-01" --name "NetworkWatcherAgentLinux" --  
publisher "Microsoft.Azure.NetworkWatcher"  
  
#Windows command  
az vm extension set --resource-group "DALANDEMO" --vm-name "Linux-01" --name "NetworkWatcherAgentWindows" --  
publisher "Microsoft.Azure.NetworkWatcher"
```

#### Option 3: Reboot your VMs

If you have auto-upgrade set to true for the Network Watcher extension, reboot your VM installation to the latest extension.

## Support

If you need more help at any point in this article, see the Network Watcher extension documentation for [Linux](#) or [Windows](#). You can also contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, file an Azure support incident. Go to the [Azure support site](#), and select **Get support**. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# InfiniBand Driver Extension for Linux

9/21/2022 • 3 minutes to read • [Edit Online](#)

This extension installs InfiniBand OFED drivers on InfiniBand and SR-IOV-enabled ('r' sizes) [H-series](#) and [N-series](#) VMs running Linux. Depending on the VM family, the extension installs the appropriate drivers for the Connect-X NIC. It does not install the InfiniBand ND drivers on the non-SR-IOV enabled [H-series](#) and [N-series](#) VMs.

Instructions on manual installation of the OFED drivers are available in [Enable InfiniBand on HPC VMs](#).

An extension is also available to install InfiniBand drivers for [Windows VMs](#).

## Prerequisites

### Operating system

This extension supports the following OS distros, depending on driver support for specific OS version.

DISTRIBUTION	VERSION	INFINIBAND NIC DRIVERS
Ubuntu	16.04 LTS, 18.04 LTS, 20.04 LTS	CX3-Pro, CX5, CX6
CentOS	7.4, 7.5, 7.6, 7.7, 7.8, 7.9, 8.1, 8.2	CX3-Pro, CX5, CX6
Red Hat Enterprise Linux	7.4, 7.5, 7.6, 7.7, 7.8, 7.9, 8.1, 8.2	CX3-Pro, CX5, CX6

For latest list of supported OS and driver versions, refer to [resources.json](#)

### Internet connectivity

The Microsoft Azure Extension for InfiniBand Drivers requires that the target VM is connected to and has access to the internet.

## Extension schema

The following JSON shows the schema for the extension.

```
{
  "name": "<myExtensionName>",
  "type": "extensions",
  "apiVersion": "2015-06-15",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <myVM>)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "InfiniBandDriverLinux",
    "typeHandlerVersion": "1.2",
    "autoUpgradeMinorVersion": true,
    "settings": {}
  }
}
```

## Properties

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2015-06-15	date
publisher	Microsoft.HpcCompute	string
type	InfiniBandDriverLinux	string
typeHandlerVersion	1.2	int

## Deployment

### Azure Resource Manager Template

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment configuration.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{  
  "name": "myExtensionName",  
  "type": "extensions",  
  "location": "[resourceGroup().location]",  
  "apiVersion": "2015-06-15",  
  "dependsOn": [  
    "[concat('Microsoft.Compute/virtualMachines/', myVM)]"  
  ],  
  "properties": {  
    "publisher": "Microsoft.HpcCompute",  
    "type": "InfiniBandDriverLinux",  
    "typeHandlerVersion": "1.2",  
    "autoUpgradeMinorVersion": true,  
    "settings": {}  
  }  
}
```

### PowerShell

```
Set-AzVMEextension  
  -ResourceGroupName "myResourceGroup" `  
  -VMName "myVM" `  
  -Location "southcentralus" `  
  -Publisher "Microsoft.HpcCompute" `  
  -ExtensionName "InfiniBandDriverLinux" `  
  -ExtensionType "InfiniBandDriverLinux" `  
  -TypeHandlerVersion 1.2 `  
  -SettingString '{ '  
}'
```

### Azure CLI

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name InfiniBandDriverLinux \
--publisher Microsoft.HpcCompute \
--version 1.2
```

## Add extension to a Virtual Machine Scale Set

The following example installs the latest version 1.2 InfiniBandDriverLinux extension on all RDMA-capable VMs in an existing virtual machine scale set named *myVMSS* deployed in the resource group named *myResourceGroup*:

```
$VMSS = Get-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "myVMSS"
Add-AzVmssExtension -VirtualMachineScaleSet $VMSS -Name "InfiniBandDriverLinux" -Publisher
"Microsoft.HpcCompute" -Type "InfiniBandDriverLinux" -TypeHandlerVersion "1.2"
Update-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "MyVMSS" -VirtualMachineScaleSet $VMSS
Update-AzVmssInstance -ResourceGroupName "myResourceGroup" -VMScaleSetName "myVMSS" -InstanceId "*"
```

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using Azure PowerShell and Azure CLI. To see the deployment state of extensions for a given VM, run the following command.

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file. Refer to this file to track the status of the installation as well as for troubleshooting any failures.

```
/var/log/azure/ib-vmext-status
```

### Exit codes

The following table describes the meaning and recommended action based on the exit codes of the extension installation process.

EXIT CODE	MEANING	POSSIBLE ACTION
0	Operation successful	
1	Incorrect usage of extension	Check execution output log
10	Linux Integration Services for Hyper-V and Azure not available or installed	Check output of lspci
11	Mellanox InfiniBand not found on this VM size	Use a <a href="#">supported VM size and OS</a>

EXIT CODE	MEANING	POSSIBLE ACTION
12	Image offer not supported	
13	VM size not supported	Use an InfiniBand-enabled ('r' size) <a href="#">H-series</a> and <a href="#">N-series</a> VM to deploy
14	Operation unsuccessful	Check execution output log

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file a support incident through the [Azure support site](#). For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

## Next steps

For more information about InfiniBand-enabled ('r' sizes), see [H-series](#) and [N-series](#) VMs.

[Learn more about Linux VMs extensions and features](#)

# InfiniBand Driver Extension for Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

This extension installs InfiniBand ND drivers (for non-SR-IOV enabled) and OFED drivers (for SR-IOV-enabled) ('r' sizes) [H-series](#) and [N-series](#) VMs running Windows. Depending on the VM family, the extension installs the appropriate drivers for the Connect-X NIC.

An extension is also available to install InfiniBand drivers for [Linux VMs](#).

## Prerequisites

### Operating system

This extension supports the following OS distros, depending on driver support for specific OS version. Note the appropriate InfiniBand NIC for the H and N-series VM sizes of interest.

DISTRIBUTION	INFINIBAND NIC DRIVERS
Windows 10	CX3-Pro, CX5, CX6
Windows Server 2019	CX3-Pro, CX5, CX6
Windows Server 2016	CX3-Pro, CX5, CX6
Windows Server 2012 R2	CX3-Pro, CX5, CX6
Windows Server 2012	CX3-Pro, CX5, CX6

For latest list of supported OS and driver versions, refer to [resources.json](#)

### Internet connectivity

The Microsoft Azure Extension for InfiniBand Drivers requires that the target VM is connected to and has access to the internet.

## Extension schema

The following JSON shows the schema for the extension.

```
{
  "name": "<myExtensionName>",
  "type": "extensions",
  "apiVersion": "2015-06-15",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <myVM>)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "InfiniBandDriverWindows",
    "typeHandlerVersion": "1.5",
    "autoUpgradeMinorVersion": true,
    "settings": {
    }
  }
}
```

## Properties

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2015-06-15	date
publisher	Microsoft.HpcCompute	string
type	InfiniBandDriverWindows	string
typeHandlerVersion	1.5	int

## Deployment

### Azure Resource Manager Template

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment configuration.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "name": "myExtensionName",
  "type": "extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "2015-06-15",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', myVM)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "InfiniBandDriverWindows",
    "typeHandlerVersion": "1.5",
    "autoUpgradeMinorVersion": true,
    "settings": {
    }
  }
}
```

## PowerShell

```
Set-AzVMExtension
  -ResourceGroupName "myResourceGroup" ` 
  -VMName "myVM" ` 
  -Location "southcentralus" ` 
  -Publisher "Microsoft.HpcCompute" ` 
  -ExtensionName "InfiniBandDriverWindows" ` 
  -ExtensionType "InfiniBandDriverWindows" ` 
  -TypeHandlerVersion 1.5 ` 
  -SettingString '{` 
}'
```

## Azure CLI

```
az vm extension set \
  --resource-group myResourceGroup \
  --vm-name myVM \
  --name InfiniBandDriverWindows \
  --publisher Microsoft.HpcCompute \
  --version 1.5
```

## Add extension to a Virtual Machine Scale Set

The following example installs the latest version 1.5 InfiniBandDriverWindows extension on all RDMA-capable VMs in an existing virtual machine scale set named *myVMSS* deployed in the resource group named *myResourceGroup*.

```
$VMSS = Get-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "myVMSS"
Add-AzVmssExtension -VirtualMachineScaleSet $VMSS -Name "InfiniBandDriverWindows" -Publisher
"Microsoft.HpcCompute" -Type "InfiniBandDriverWindows" -TypeHandlerVersion "1.5"
Update-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "MyVMSS" -VirtualMachineScaleSet $VMSS
Update-AzVmssInstance -ResourceGroupName "myResourceGroup" -VMScaleSetName "myVMSS" -InstanceId "*"
```

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using Azure PowerShell and Azure CLI. To see the deployment state of extensions for a given VM, run the following command.

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file. Refer to this file to track the status of the installation as well as for troubleshooting any failures.

```
C:\WindowsAzure\Logs\Plugins\Microsoft.HpcCompute.InfiniBandDriverWindows\
```

## Exit codes

The following table describes the meaning and recommended action based on the exit codes of the extension installation process.

ERROR CODE	MEANING	POSSIBLE ACTION
0	Operation successful	
3010	Operation successful. Reboot required.	
100	Operation not supported or could not be completed.	Possible causes: PowerShell version not supported, VM size is not an InfiniBand-enabled VM, Failure downloading data. Check the log files to determine cause of error.
240, 840	Operation timeout.	Retry operation.
-1	Exception occurred.	Check the log files to determine cause of exception.

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file a support incident through the [Azure support site](#). For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

## Next steps

For more information about InfiniBand-enabled ('r' sizes), see [H-series](#) and [N-series](#) VMs.

[Learn more about Linux VMs extensions and features](#)

# NVIDIA GPU Driver Extension for Linux

9/21/2022 • 4 minutes to read • [Edit Online](#)

This extension installs NVIDIA GPU drivers on Linux N-series virtual machines (VMs). Depending on the VM family, the extension installs CUDA or GRID drivers. When you install NVIDIA drivers by using this extension, you're accepting and agreeing to the terms of the [NVIDIA End-User License Agreement](#). During the installation process, the VM might reboot to complete the driver setup.

Instructions on manual installation of the drivers and the current supported versions are available. For more information, see [Azure N-series GPU driver setup for Linux](#). An extension is also available to install NVIDIA GPU drivers on [Windows N-series VMs](#).

## Prerequisites

### Operating system

This extension supports the following OS distros, depending on driver support for the specific OS version:

DISTRIBUTION	VERSION
Linux: Ubuntu	16.04 LTS, 18.04 LTS, 20.04 LTS
Linux: Red Hat Enterprise Linux	7.3, 7.4, 7.5, 7.6, 7.7, 7.8
Linux: CentOS	7.3, 7.4, 7.5, 7.6, 7.7, 7.8

#### NOTE

The latest supported CUDA drivers for NC-series VMs are currently 470.82.01. Later driver versions aren't supported on the K80 cards in NC. While the extension is being updated with this end of support for NC, install CUDA drivers manually for K80 cards on the NC-series.

### Internet connectivity

The Microsoft Azure Extension for NVIDIA GPU Drivers requires that the target VM is connected to the internet and has access.

## Extension schema

The following JSON shows the schema for the extension:

```
{
  "name": "<myExtensionName>",
  "type": "extensions",
  "apiVersion": "2015-06-15",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <myVM>)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "NvidiaGpuDriverLinux",
    "typeHandlerVersion": "1.6",
    "autoUpgradeMinorVersion": true,
    "settings": {
    }
  }
}
```

## Properties

NAME	VALUE/EXAMPLE	DATA TYPE
apiVersion	2015-06-15	date
publisher	Microsoft.HpcCompute	string
type	NvidiaGpuDriverLinux	string
typeHandlerVersion	1.6	int

## Settings

All settings are optional. The default behavior is to not update the kernel if not required for driver installation and install the latest supported driver and the CUDA toolkit (as applicable).

NAME	DESCRIPTION	DEFAULT VALUE	VALID VALUES	DATA TYPE
updateOS	Update the kernel even if not required for driver installation.	false	true, false	boolean
driverVersion	NV: GRID driver version. NC/ND: CUDA toolkit version. The latest drivers for the chosen CUDA are installed automatically.	latest	<a href="#">List</a> of supported driver versions	string
installCUDA	Install CUDA toolkit. Only relevant for NC/ND series VMs.	true	true, false	boolean

## Deployment

### Azure portal

You can deploy Azure NVIDIA VM extensions in the Azure portal.

1. In a browser, go to the [Azure portal](#).
2. Go to the virtual machine on which you want to install the driver.
3. On the left menu, select **Extensions**.

**MyTestVM - Microsoft Azure**

**Microsoft Azure (Preview)** [Report a bug](#) [Search resources, services, and docs \(G+\)](#)

Home > MyTestVM Virtual machine

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings Networking Connect Disks Size Security Advisor recommendations Extensions + applications (highlighted with a red box) Continuous delivery Availability + scaling Configuration Identity

**Essentials**

Resource group (Move) :	mytestvm
Status	: Running
Location	: West US 2
Subscription (Move)	: se1personal
Subscription ID	:
Tags (Edit)	: Click here to add tags

Properties Monitoring Capabilities (7) Recommendations (2) Tutorials

**Virtual machine**

Computer name	MyTestVM
Health state	-
Operating system	Linux
Publisher	Canonical
Offer	UbuntuServer
Plan	18_04-lts-gen2
VM generation	V2
Host group	None

**Networking**

Public IP address	-
Public IP address (IPv6)	-
Private IP address	-
Private IP address (IPv6)	-
Virtual network/subnet	mytestvm-vnet/default
DNS name	Configure

**Size**

4. Select **Add**.

**Microsoft Azure (Preview)** [Report a bug](#) [Search resources, services, and docs \(G+\)](#)

Home > MyTestVM

**MyTestVM | Extensions + applications**

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings Networking Connect Disks Size Security Advisor recommendations Extensions + applications (highlighted with a red box) Continuous delivery Availability + scaling

**Extensions** **VM Applications**

+ Add Refresh

Search to filter items...

Name	Type
OMSAgentForLinux	Microsoft.EnterpriseCloud.Monitoring.OmsAgentForLinux

5. Scroll to find and select **NVIDIA GPU Driver Extension**, and then select **Next**.

[Home](#) > [MyTestVM](#) >

## Install an Extension



### Kaspersky Hybrid Cloud Security Agent

Kaspersky Lab

Linux Workload Security with unified orchestration, automated discovery and provisioning through native Azure API integration



### Network Watcher Agent for Linux

Microsoft Corp.

Azure Network Watcher is a network performance monitoring, diagnostic and analytics service that enables you to monitor your network in Azure



### New Relic Infrastructure

New Relic

New Relic Infrastructure provides flexible, dynamic monitoring of your entire infrastructure.



### NVIDIA GPU Driver Extension

Microsoft Corp.

Microsoft Azure Extension for NVIDIA GPU Drivers



### Rapid7 Insight Agent

Rapid7

The Rapid7 Insight Agent connects your Azure VMs to solutions on the Rapid7 Insight platform.



### SentinelOne Linux Extension

SentinelOne

Provision SentinelOne Linux Agent for best of class threat detection, prevention, and remediation.



### Stackify Retrace Linux Agent

Stackify

This will install the Linux Agent for Retrace.

[Next](#)

6. Select **Review + create**, and select **Create**. Wait a few minutes for the driver to deploy.

[Home](#) > [MyTestVM](#) > [Install an Extension](#) >

## Configure NVIDIA GPU Driver Extension Extension ...

[Create](#)    [Review + create](#)

**i** This extension installs NVIDIA drivers on Linux N-series VMs. Depending on the VM family, the extension installs CUDA or GRID drivers. When you install NVIDIA drivers using this extension, you are accepting and agreeing to the terms of the NVIDIA End User License Agreement. During the installation process, your virtual machine may reboot to complete the driver setup.

[Review + create](#)

&lt; Previous

Next : Review + create &gt;

7. Verify that the extension was added to the list of installed extensions.

MyTestVM | Extensions + applications

Name	Type	Version	Status
NvidiaGpuDriverLinux	Microsoft.HpcCompute.NvidiaG	1.*	Provisioning succeeded
OMSAgentForLinux	Microsoft.EnterpriseCloud.Monitoring	1.*	Provisioning succeeded

## Azure Resource Manager template

You can use Azure Resource Manager templates to deploy Azure VM extensions. Templates are ideal when you deploy one or more virtual machines that require post-deployment configuration.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the extension is nested inside the virtual machine resource. When the extension resource is nested, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{  
  "name": "myExtensionName",  
  "type": "extensions",  
  "location": "[resourceGroup().location]",  
  "apiVersion": "2015-06-15",  
  "dependsOn": [  
    "[concat('Microsoft.Compute/virtualMachines/', myVM)]"  
  ],  
  "properties": {  
    "publisher": "Microsoft.HpcCompute",  
    "type": "NvidiaGpuDriverLinux",  
    "typeHandlerVersion": "1.6",  
    "autoUpgradeMinorVersion": true,  
    "settings": {}  
  }  
}
```

## PowerShell

```
Set-AzVMExtension
  -ResourceGroupName "myResourceGroup" ` 
  -VMName "myVM" ` 
  -Location "southcentralus" ` 
  -Publisher "Microsoft.HpcCompute" ` 
  -ExtensionName "NvidiaGpuDriverLinux" ` 
  -ExtensionType "NvidiaGpuDriverLinux" ` 
  -TypeHandlerVersion 1.6 ` 
  -SettingString '{ ` 
}'
```

## Azure CLI

The following example mirrors the preceding Resource Manager and PowerShell examples:

```
az vm extension set \
  --resource-group myResourceGroup \
  --vm-name myVM \
  --name NvidiaGpuDriverLinux \
  --publisher Microsoft.HpcCompute \
  --version 1.6
```

The following example also adds two optional custom settings as an example for nondefault driver installation. Specifically, it updates the OS kernel to the latest and installs a specific CUDA toolkit version driver. Again, note the `--settings` are optional and default. Updating the kernel might increase the extension installation times. Also, choosing a specific (older) CUDA toolkit version might not always be compatible with newer kernels.

```
az vm extension set \
  --resource-group myResourceGroup \
  --vm-name myVM \
  --name NvidiaGpuDriverLinux \
  --publisher Microsoft.HpcCompute \
  --version 1.6 \
  --settings '{ \
    "updateOS": true, \
    "driverVersion": "10.0.130" \
}'
```

## Troubleshoot and support

### Troubleshoot

You can retrieve data about the state of extension deployments from the Azure portal and by using Azure PowerShell and the Azure CLI. To see the deployment state of extensions for a given VM, run the following command:

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file. Refer to this file to track the status of any long-running installation and for troubleshooting any failures.

```
/var/log/azure/nvidia-vmext-status
```

## Exit codes

EXIT CODE	MEANING	POSSIBLE ACTION
0	Operation successful	
1	Incorrect usage of extension	Check the execution output log.
10	Linux Integration Services for Hyper-V and Azure not available or installed	Check the output of <code>lspci</code> .
11	NVIDIA GPU not found on this VM size	Use a <a href="#">supported VM size and OS</a> .
12	Image offer not supported	
13	VM size not supported	Use an N-series VM to deploy.
14	Operation unsuccessful	Check the execution output log.

## Support

If you need more help at any point in this article, contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to [Azure support](#) and select **Get support**. For information about using Azure support, read the [Azure support FAQ](#).

## Next steps

- For more information about extensions, see [Virtual machine extensions and features for Linux](#).
- For more information about N-series VMs, see [GPU optimized virtual machine sizes](#).

# NVIDIA GPU Driver Extension for Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

This extension installs NVIDIA GPU drivers on Windows N-series virtual machines (VMs). Depending on the VM family, the extension installs CUDA or GRID drivers. When you install NVIDIA drivers by using this extension, you're accepting and agreeing to the terms of the [NVIDIA End-User License Agreement](#). During the installation process, the VM might reboot to complete the driver setup.

Instructions on manual installation of the drivers and the current supported versions are available. For more information, see [Azure N-series NVIDIA GPU driver setup for Windows](#). An extension is also available to install NVIDIA GPU drivers on [Linux N-series VMs](#).

## Prerequisites

### Operating system

This extension supports the following OSs:

DISTRIBUTION	VERSION
Windows 10	Core
Windows Server 2019	Core
Windows Server 2016	Core
Windows Server 2012 R2	Core

### Internet connectivity

The Microsoft Azure Extension for NVIDIA GPU Drivers requires that the target VM is connected to the internet and has access.

## Extension schema

The following JSON shows the schema for the extension:

```
{
  "name": "<myExtensionName>",
  "type": "extensions",
  "apiVersion": "2015-06-15",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <myVM>)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "NvidiaGpuDriverWindows",
    "typeHandlerVersion": "1.4",
    "autoUpgradeMinorVersion": true,
    "settings": {}
  }
}
```

## Properties

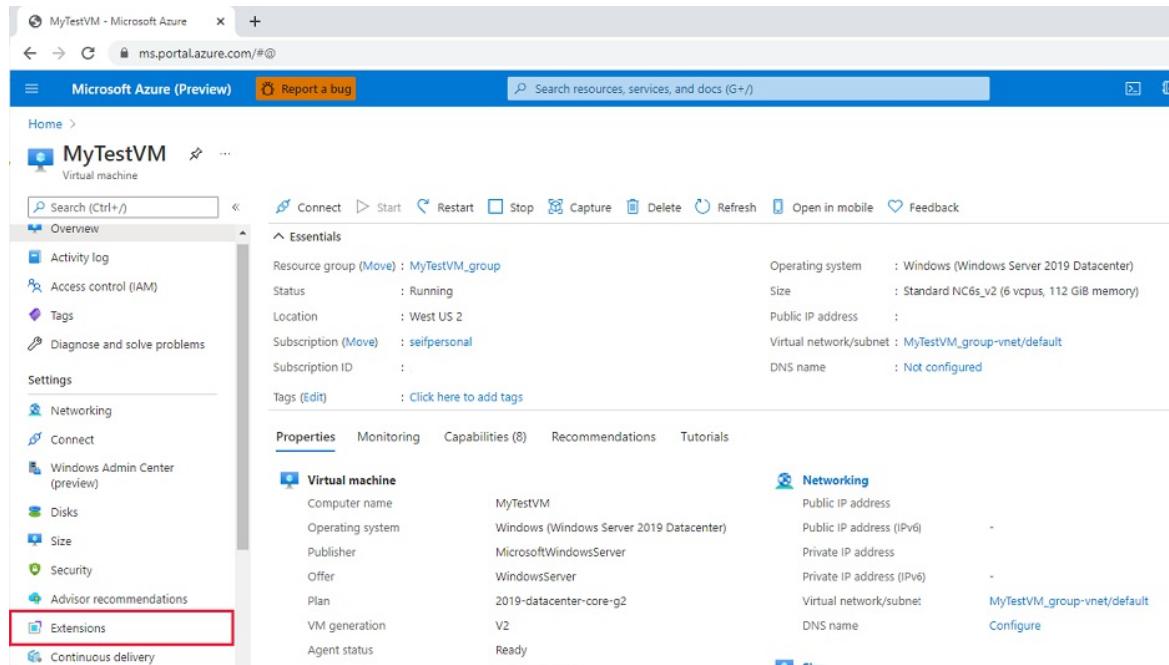
Name	Value/Example	Data Type
apiVersion	2015-06-15	date
publisher	Microsoft.HpcCompute	string
type	NvidiaGpuDriverWindows	string
typeHandlerVersion	1.4	int

## Deployment

### Azure portal

You can deploy Azure NVIDIA VM extensions in the Azure portal.

1. In a browser, go to the [Azure portal](#).
2. Go to the virtual machine on which you want to install the driver.
3. On the left menu, select **Extensions**.



The screenshot shows the Azure portal interface for a virtual machine named 'MyTestVM'. The left sidebar has a 'Extensions' link highlighted with a red box. The main content area displays the 'Essentials' and 'Virtual machine' sections. The 'Virtual machine' section includes details like Computer name (MyTestVM), Operating system (Windows (Windows Server 2019 Datacenter)), and Publisher (MicrosoftWindowsServer). The 'Networking' section shows Public IP address and Private IP address details. The 'Properties' tab is selected at the bottom.

4. Select **Add**.

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > MyTestVM

## MyTestVM | Extensions

Virtual machine

+ Add

Search to filter items...

Name	Type
MMAExtension	Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent

Search (Ctrl+ /)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Networking

Connect

Windows Admin Center (preview)

Disk

Size

Security

Advisor recommendations

Extensions

Continuous delivery

Availability + scaling

Configuration

5. Scroll to find and select NVIDIA GPU Driver Extension, and then select Next.

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > MyTestVM >

## Install an Extension

**Datadog Agent** Datadog Inc. Datadog monitoring agent will enable you collect detailed information on applications running inside your VM instances.

**DxEnterprise for Windows** DH2i Company Create SQL Server FCIs and Availability Groups without WSFC/Pacemaker or networking complexity.

**Dynatrace OneAgent** Dynatrace Dynatrace OneAgent for Windows.

**ESET File Security** ESET ESET File Security extension for Microsoft Azure.

**HPE Security Fortify Application Defender** HPE Security Fortify Application Defender Agent

**Kaspersky Hybrid Cloud Security Agent** Kaspersky Lab Windows workload security with unified orchestration, automated discovery and provisioning through native Azure API integration.

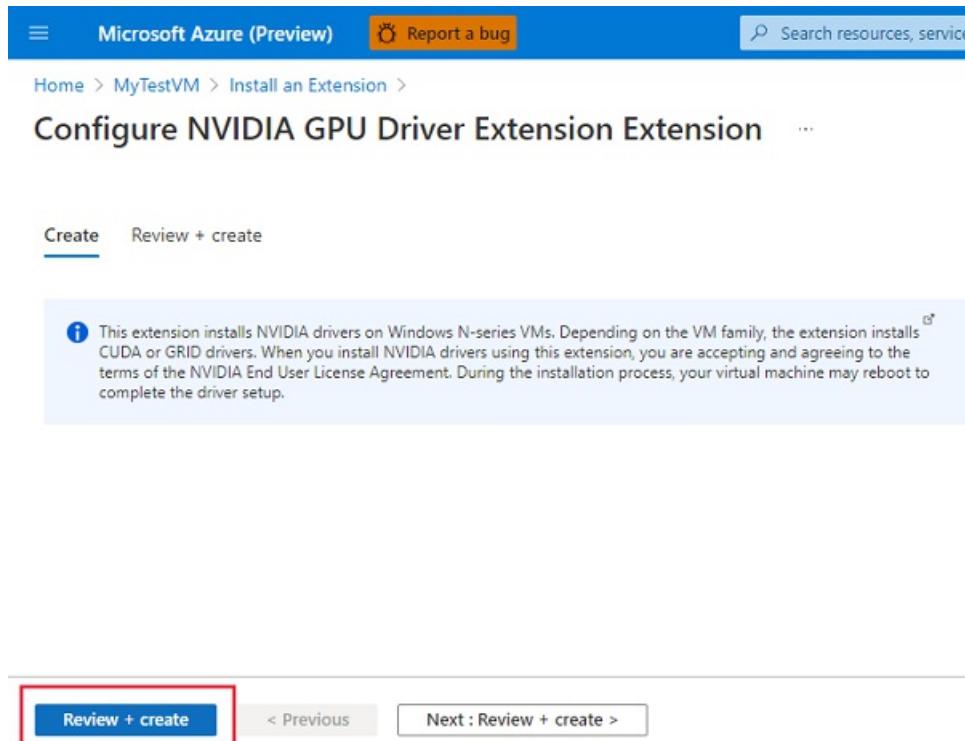
**Microsoft Antimalware** Microsoft Corp. Microsoft Antimalware for Azure Virtual Machines.

**NVIDIA GPU Driver Extension** Microsoft Corp. Microsoft Azure Extension for NVIDIA GPU Drivers.

Load more

Next

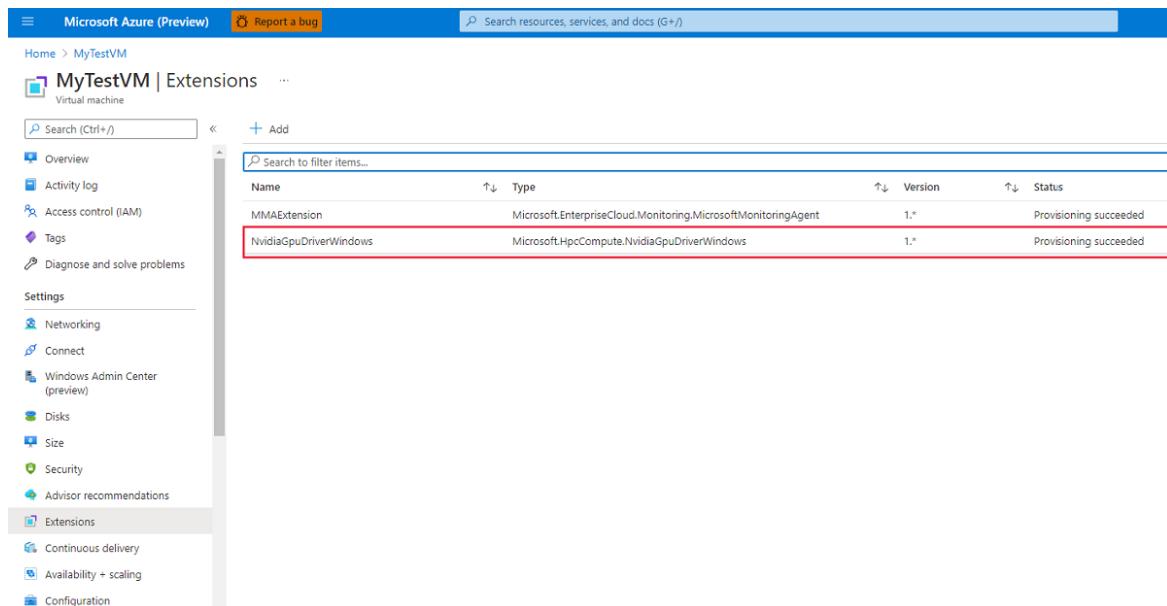
6. Select **Review + create**, and select **Create**. Wait a few minutes for the driver to deploy.



This extension installs NVIDIA drivers on Windows N-series VMs. Depending on the VM family, the extension installs CUDA or GRID drivers. When you install NVIDIA drivers using this extension, you are accepting and agreeing to the terms of the NVIDIA End User License Agreement. During the installation process, your virtual machine may reboot to complete the driver setup.

**Review + create**

7. Verify that the extension was added to the list of installed extensions.



Name	Type	Version	Status
MMAExtension	Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent	1.0	Provisioning succeeded
NvidiaGpuDriverWindows	Microsoft.HpcCompute.NvidiaGpuDriverWindows	1.0	Provisioning succeeded

### Azure Resource Manager template

You can use Azure Resource Manager templates to deploy Azure VM extensions. Templates are ideal when you deploy one or more virtual machines that require post-deployment configuration.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the extension is nested inside the virtual machine resource. When the extension resource is nested, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "name": "myExtensionName",
  "type": "extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "2015-06-15",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', myVM)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "NvidiaGpuDriverWindows",
    "typeHandlerVersion": "1.4",
    "autoUpgradeMinorVersion": true,
    "settings": {
    }
  }
}
```

## PowerShell

```
Set-AzVMExtension
-ResourceGroupName "myResourceGroup" ` 
-VMName "myVM" ` 
-Location "southcentralus" ` 
-Publisher "Microsoft.HpcCompute" ` 
-ExtensionName "NvidiaGpuDriverWindows" ` 
-ExtensionType "NvidiaGpuDriverWindows" ` 
-TypeHandlerVersion 1.4 ` 
-SettingString '{ ' ` 
}'
```

## Azure CLI

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name NvidiaGpuDriverWindows \
--publisher Microsoft.HpcCompute \
--version 1.4 \
--settings '{ ' \
}'
```

## Troubleshoot and support

### Troubleshoot

You can retrieve data about the state of extension deployments from the Azure portal and by using Azure PowerShell and the Azure CLI. To see the deployment state of extensions for a given VM, run the following command:

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following directory:

C:\WindowsAzure\Logs\Plugins\Microsoft.HpcCompute.NvidiaGpuDriverWindows\

## Error codes

ERROR CODE	MEANING	POSSIBLE ACTION
0	Operation successful.	
1	Operation successful. Reboot required.	
100	Operation not supported or couldn't be completed.	Possible causes are that the PowerShell version isn't supported, the VM size isn't an N-series VM, or a failure occurred in downloading data. Check the log files to determine the cause of the error.
240, 840	Operation timeout.	Retry operation.
-1	Exception occurred.	Check the log files to determine the cause of the exception.
-5x	Operation interrupted due to pending reboot.	Reboot VM. Installation continues after the reboot. Uninstall should be invoked manually.

## Support

If you need more help at any point in this article, contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to [Azure support](#) and select **Get support**. For information about using Azure support, read the [Azure support FAQ](#).

## Next steps

- For more information about extensions, see [Virtual machine extensions and features for Windows](#).
- For more information about N-series VMs, see [GPU optimized virtual machine sizes](#).

# AMD GPU Driver Extension for Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

This article provides an overview of the virtual machine (VM) extension to deploy AMD GPU drivers on Windows NVv4-series VMs. When you install AMD drivers by using this extension, you're accepting and agreeing to the terms of the [AMD End-User License Agreement](#). During the installation process, the VM might reboot to complete the driver setup.

Instructions on manual installation of the drivers and the current supported versions are available. For more information, see [Azure N-series AMD GPU driver setup for Windows](#).

## Prerequisites

### Operating system

This extension supports the following OSs:

DISTRIBUTION	VERSION
Windows 10 EMS	Build 1909
Windows 10	Build 1909
Windows Server 2016	Core
Windows Server 2019	Core

### Internet connectivity

The Microsoft Azure Extension for AMD GPU Drivers requires that the target VM is connected to the internet and has access.

## Extension schema

The following JSON shows the schema for the extension:

```
{
  "name": "<myExtensionName>",
  "type": "extensions",
  "apiVersion": "2015-06-15",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <myVM>)]"
  ],
  "properties": {
    "publisher": "Microsoft.HpcCompute",
    "type": "AmdGpuDriverWindows",
    "typeHandlerVersion": "1.1",
    "autoUpgradeMinorVersion": true,
    "settings": {}
  }
}
```

## Properties

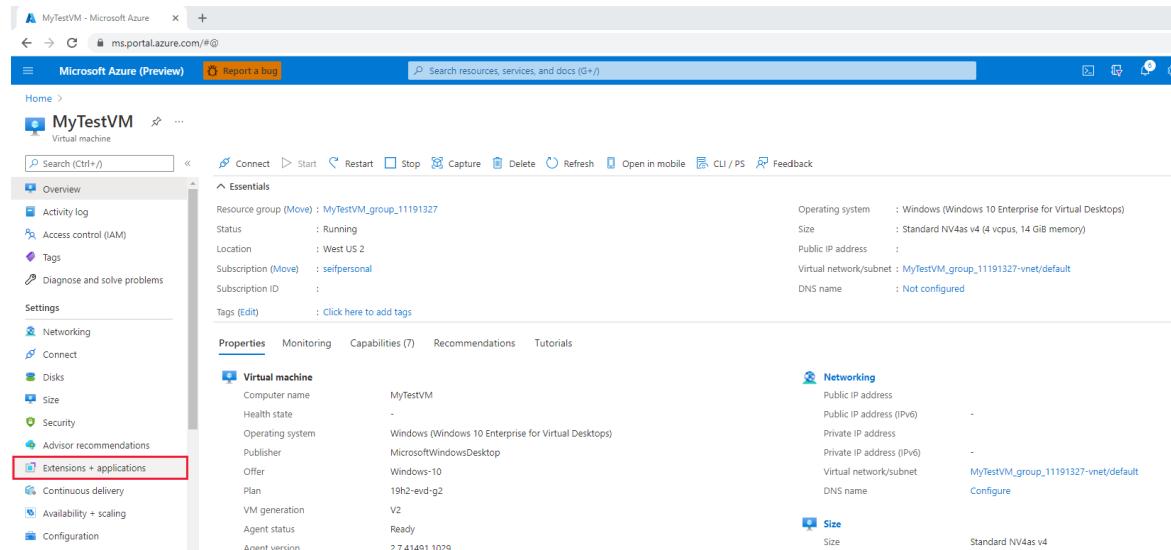
Name	Value/Example	Data Type
apiVersion	2015-06-15	date
publisher	Microsoft.HpcCompute	string
type	AmdGpuDriverWindows	string
typeHandlerVersion	1.1	int

## Deployment

### Azure portal

You can deploy Azure AMD VM extensions in the Azure portal.

1. In a browser, go to the [Azure portal](#).
2. Go to the virtual machine on which you want to install the driver.
3. On the left menu, select **Extensions**.



The screenshot shows the Azure portal interface for a virtual machine named 'MyTestVM'. The left sidebar shows navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Disks, Size, Security, Advisor recommendations, and Extensions + applications. The 'Extensions + applications' link is highlighted with a red box. The main content area displays the virtual machine's properties under the 'Virtual machine' tab, including Computer name (MyTestVM), Health state, Operating system (Windows 10 Enterprise for Virtual Desktops), Publisher (MicrosoftWindowsDesktop), Offer (Windows-10), Plan (19h2-evd-g2), VM generation (V2), Agent status (Ready), and Agent version (2.7.41491.1029). To the right, there are sections for Networking (Public IP address, Private IP address, Virtual network/subnet, DNS name) and Size (Size: Standard NV4as v4). At the top, there are buttons for Connect, Start, Stop, Capture, Delete, Refresh, Open in mobile, CLI / PS, and Feedback.

4. Select **Add**.

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > MyTestVM

## MyTestVM | Extensions + applications

Virtual machine

Search (Ctrl+ /) Extensions VM Applications

+ Add Refresh

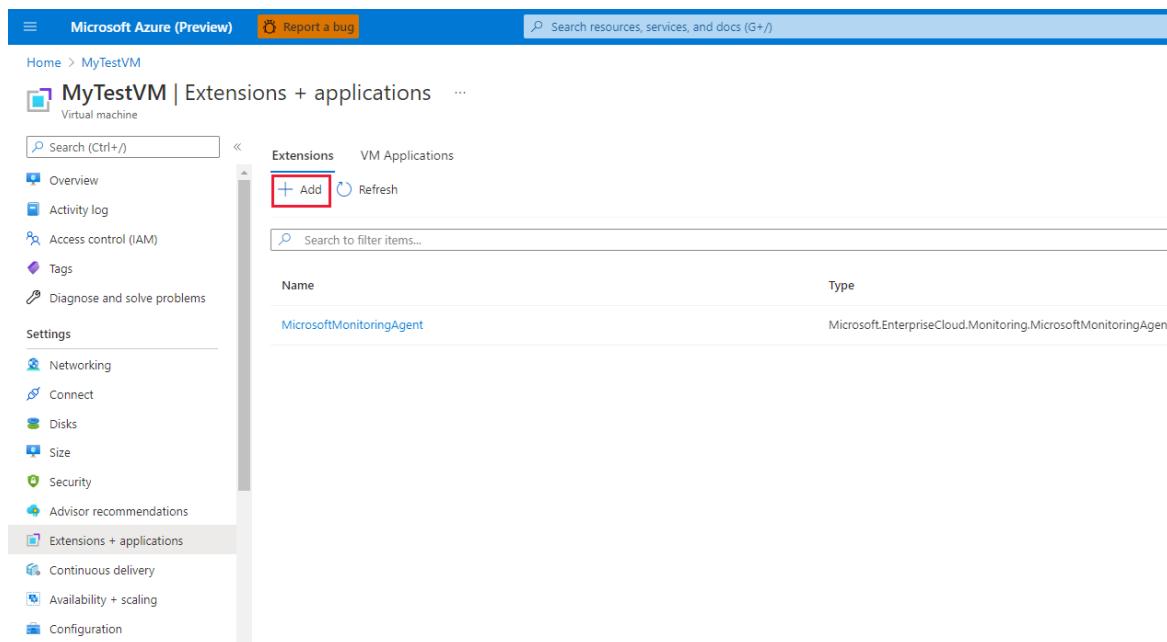
Search to filter items...

Name	Type
MicrosoftMonitoringAgent	Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings Networking Connect Disks Size Security Advisor recommendations

Extensions + applications Continuous delivery Availability + scaling Configuration



5. Scroll to find and select AMD GPU Driver Extension, and then select **Next**.

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > MyTestVM >

## Install an Extension

Search

**Acronis Backup**  
Acronis, Inc.

The most complete, cost-effective and easy-to-manage hybrid local and cloud backup solution available

**Agent for Cloud Workload Protection (Windows)**  
Symantec Corp.

Symantec Cloud Workload Protection provides strong security for servers with application protection, intrusion detection/prevention, real-time Anti-

**Agent for Windows Server Monitoring**  
Site24x7

Ensure your Windows server and applications are up and running with our performance data and on-time alerts

**AMD GPU Driver Extension**  
Microsoft Corp.

Microsoft Azure Extension for AMD GPU Drivers

**APM Insight .NET Agent**  
Site24x7

Get real time, comprehensive data on all your .NET web applications using Site24x7 APM Insight.

**Application Insights Agent (.NET Preview)**  
Microsoft Corp.

Application Insights Agent (.NET Preview)

**Azure AD based Windows Login**  
Microsoft Corp.

This extension configures your Windows VM for Azure AD based login.

**Azure Performance Diagnostics**  
Microsoft Corp.

Azure Performance Diagnostics extension helps monitor and troubleshoot various performance issues on the VM

**Azure Pipelines Agent**  
Microsoft Corp.

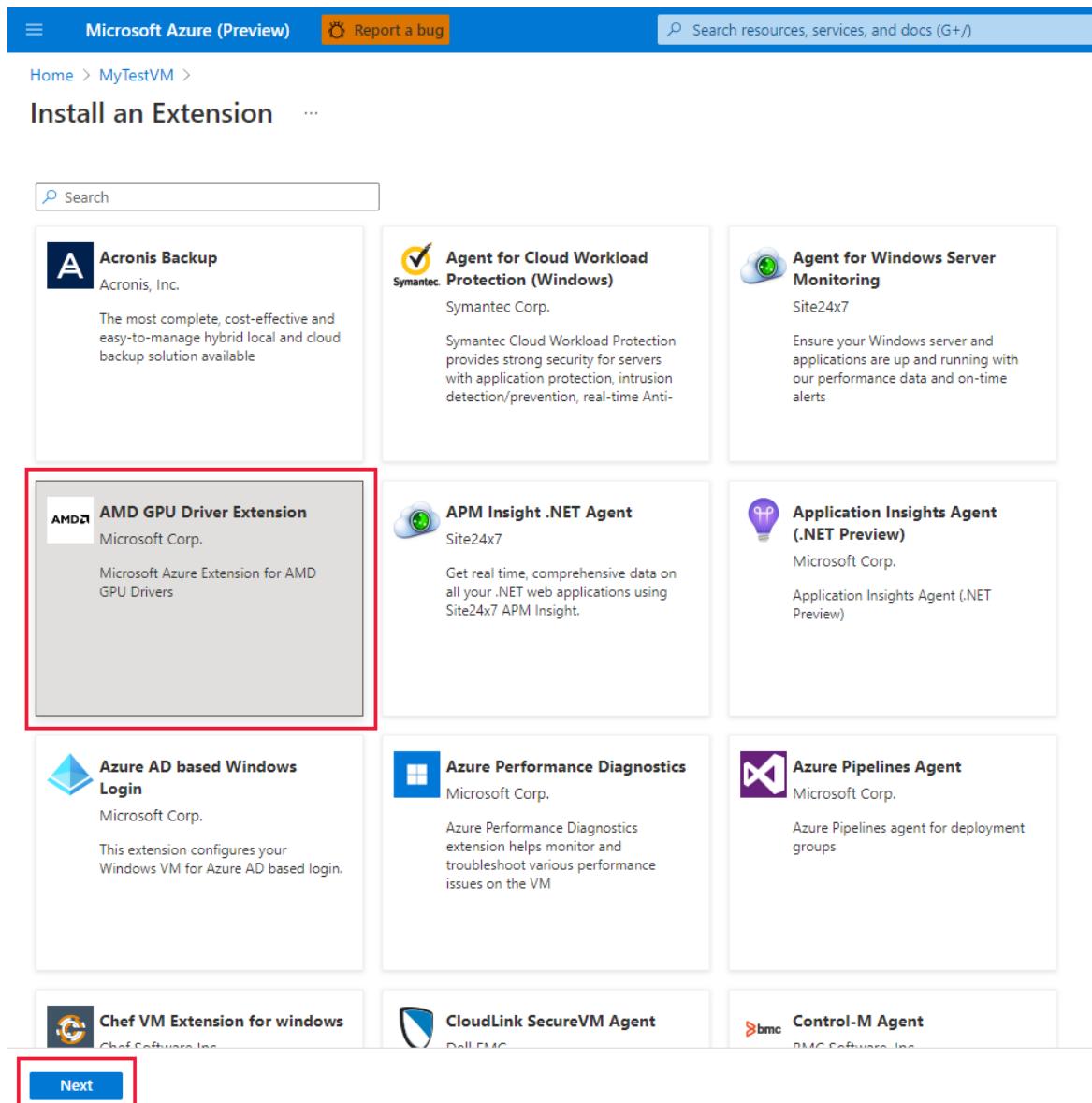
Azure Pipelines agent for deployment groups

**Chef VM Extension for windows**  
Chef Software, Inc.

**CloudLink SecureVM Agent**  
Dell EMC

**Control-M Agent**  
BMC Software, Inc.

**Next**



6. Select **Review + create**, and select **Create**. Wait a few minutes for the driver to deploy.

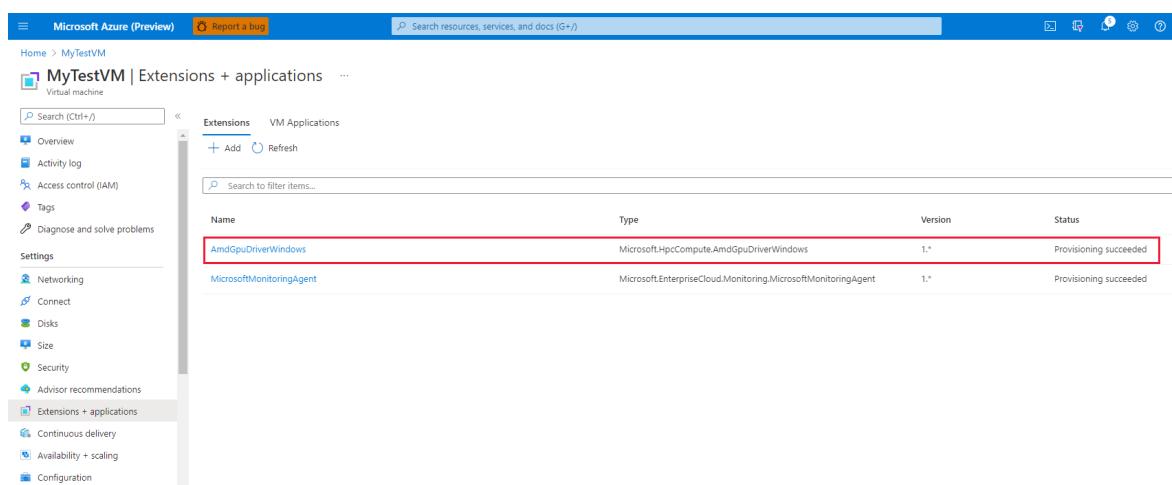
## Configure AMD GPU Driver Extension Extension

[Create](#)    [Review + create](#)

**i** This extension installs AMD GPU drivers on Windows NV v4 series VMs. When you install AMD GPU drivers using this extension, you are accepting and agreeing to the terms of the AMD End User License Agreement. During the installation process, your virtual machine may reboot to complete the driver setup.

[Review + create](#)
< Previous
Next : Review + create >

7. Verify that the extension was added to the list of installed extensions.



Name	Type	Version	Status
AmdGpuDriverWindows	Microsoft.HpcCompute.AmdGpuDriverWindows	1.*	Provisioning succeeded
MicrosoftMonitoringAgent	Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent	1.*	Provisioning succeeded

### Azure Resource Manager template

You can use Azure Resource Manager templates to deploy Azure VM extensions. Templates are ideal when you

deploy one or more virtual machines that require post-deployment configuration.

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the extension is nested inside the virtual machine resource. When the extension resource is nested, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{  
  "name": "myExtensionName",  
  "type": "extensions",  
  "location": "[resourceGroup().location]",  
  "apiVersion": "2015-06-15",  
  "dependsOn": [  
    "[concat('Microsoft.Compute/virtualMachines/', myVM)]"  
  ],  
  "properties": {  
    "publisher": "Microsoft.HpcCompute",  
    "type": "AmdGpuDriverWindows",  
    "typeHandlerVersion": "1.1",  
    "autoUpgradeMinorVersion": true,  
    "settings": {}  
  }  
}
```

## PowerShell

```
Set-AzVMExtension  
  -ResourceGroupName "myResourceGroup" `  
  -VMName "myVM" `  
  -Location "southcentralus" `  
  -Publisher "Microsoft.HpcCompute" `  
  -ExtensionName "AmdGpuDriverWindows" `  
  -ExtensionType "AmdGpuDriverWindows" `  
  -TypeHandlerVersion 1.1 `  
  -SettingString '{`  
}'
```

## Azure CLI

```
az vm extension set `  
  --resource-group myResourceGroup `  
  --vm-name myVM `  
  --name AmdGpuDriverWindows `  
  --publisher Microsoft.HpcCompute `  
  --version 1.1 `  
  --settings '{`  
}'
```

## Troubleshoot and support

### Troubleshoot

You can retrieve data about the state of extension deployments from the Azure portal and by using Azure PowerShell and the Azure CLI. To see the deployment state of extensions for a given VM, run the following command:

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following directory:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.HpcCompute.AmdGpuDriverMicrosoft\
```

## Error codes

ERROR CODE	MEANING	POSSIBLE ACTION
0	Operation successful.	
1	Operation successful. Reboot required.	
100	Operation not supported or couldn't be completed.	Possible causes are that the PowerShell version isn't supported, the VM size isn't an N-series VM, and a failure occurred in downloading data. Check the log files to determine the cause of the error.
240, 840	Operation timeout.	Retry operation.
-1	Exception occurred.	Check the log files to determine the cause of the exception.
-5x	Operation interrupted due to pending reboot.	Reboot VM. Installation continues after the reboot. Uninstall should be invoked manually.

## Support

If you need more help at any point in this article, contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to [Azure support](#) and select **Get support**. For information about using Azure support, read the [Azure support FAQ](#).

## Next steps

- For more information about extensions, see [Virtual machine extensions and features for Windows](#).
- For more information about N-series VMs, see [GPU optimized virtual machine sizes](#).

# Manage the Azure Monitor agent

9/21/2022 • 11 minutes to read • [Edit Online](#)

This article provides the different options currently available to install, uninstall, and update the [Azure Monitor agent](#). This agent extension can be installed on Azure virtual machines, scale sets, and Azure Arc-enabled servers. It also lists the options to create [associations with data collection rules](#) that define which data the agent should collect. Installing, upgrading, or uninstalling the Azure Monitor agent won't require you to restart your server.

## Virtual machine extension details

The Azure Monitor agent is implemented as an [Azure VM extension](#) with the details in the following table. You can install it by using any of the methods to install virtual machine extensions including the methods described in this article.

PROPERTY	WINDOWS	LINUX
Publisher	Microsoft.Azure.Monitor	Microsoft.Azure.Monitor
Type	AzureMonitorWindowsAgent	AzureMonitorLinuxAgent
TypeHandlerVersion	See <a href="#">Azure Monitor agent extension versions</a>	<a href="#">Azure Monitor agent extension versions</a>

## Extension versions

View [Azure Monitor agent extension versions](#).

## Prerequisites

The following prerequisites must be met prior to installing the Azure Monitor agent.

- **Permissions:** For methods other than using the Azure portal, you must have the following role assignments to install the agent:

BUILT-IN ROLE	SCOPES	REASON
• <a href="#">Virtual Machine Contributor</a> • <a href="#">Azure Connected Machine Resource Administrator</a>	• Virtual machines, scale sets, • Azure Arc-enabled servers	To deploy the agent
Any role that includes the action <code>Microsoft.Resources/deployments/*</code>	• Subscription and/or • Resource group and/or	To deploy Azure Resource Manager templates

- **Non-Azure:** To install the agent on physical servers and virtual machines hosted *outside* of Azure (that is, on-premises) or in other clouds, you must [install the Azure Arc Connected Machine agent](#) first, at no added cost.
- **Authentication:** [Managed identity](#) must be enabled on Azure virtual machines. Both user-assigned and

system-assigned managed identities are supported.

- **User-assigned:** This managed identity is recommended for large-scale deployments, configurable via [built-in Azure policies](#). You can create a user-assigned managed identity once and share it across multiple VMs, which means it's more scalable than a system-assigned managed identity. If you use a user-assigned managed identity, you must pass the managed identity details to the Azure Monitor agent via extension settings:

```
{  
  "authentication": {  
    "managedIdentity": {  
      "identifier-name": "mi_res_id" or "object_id" or "client_id",  
      "identifier-value": "<resource-id-of-uai>" or "<guid-object-or-client-id>"  
    }  
  }  
}
```

We recommend that you use `mi_res_id` as the `identifier-name`. The following sample commands only show usage with `mi_res_id` for the sake of brevity. For more information on `mi_res_id`, `object_id`, and `client_id`, see the [Managed identity documentation](#).

- **System-assigned:** This managed identity is suited for initial testing or small deployments. When used at scale, for example, for all VMs in a subscription, it results in a substantial number of identities created (and deleted) in Azure Active Directory. To avoid this churn of identities, use user-assigned managed identities instead. *For Azure Arc-enabled servers, system-assigned managed identity is enabled automatically as soon as you install the Azure Arc agent. It's the only supported type for Azure Arc-enabled servers.*
- **Not required for Azure Arc-enabled servers:** The system identity is enabled automatically if the agent is installed via [creating and assigning a data collection rule by using the Azure portal](#).
- **Networking:** If you use network firewalls, the [AzureResourceManager service tag](#) must be enabled on the virtual network for the virtual machine. The virtual machine must also have access to the following HTTPS endpoints:
  - `global.handler.control.monitor.azure.com`
  - `<virtual-machine-region-name>.handler.control.monitor.azure.com` (example: `westus.handler.control.azure.com`)
  - `<log-analytics-workspace-id>.ods.opinsights.azure.com` (example: `12345a01-b1cd-1234-e1f2-1234567g8h99.ods.opinsights.azure.com`)  
(If you use private links on the agent, you must also add the [dce endpoints](#)).

#### NOTE

This article only pertains to agent installation or management. After you install the agent, you must review the next article to [configure data collection rules and associate them with the machines](#) with agents installed. *The Azure Monitor agents can't function without being associated with data collection rules.*

## Use the Azure portal

Follow these instructions to use the Azure portal.

### Install

To install the Azure Monitor agent by using the Azure portal, follow the process to [create a data collection rule](#) in the Azure portal. This process creates the rule, associates it to the selected resources, and installs the Azure

Monitor agent on them if it's not already installed.

## Uninstall

To uninstall the Azure Monitor agent by using the Azure portal, go to your virtual machine, scale set, or Azure Arc-enabled server. Select the **Extensions** tab and select **AzureMonitorWindowsAgent** or **AzureMonitorLinuxAgent**. In the dialog that opens, select **Uninstall**.

## Update

To perform a one-time update of the agent, you must first uninstall the existing agent version. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature. Go to your virtual machine or scale set, select the **Extensions** tab and select **AzureMonitorWindowsAgent** or **AzureMonitorLinuxAgent**. In the dialog that opens, select **Enable automatic upgrade**.

# Use Resource Manager templates

Follow these instructions to use Azure Resource Manager templates.

## Install

You can use Resource Manager templates to install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers and to create an association with data collection rules. You must create any data collection rule prior to creating the association.

Get sample templates for installing the agent and creating the association from the following resources:

- [Template to install Azure Monitor agent \(Azure and Azure Arc\)](#)
- [Template to create association with data collection rule](#)

Install the templates by using [any deployment method for Resource Manager templates](#), such as the following commands.

- [PowerShell](#)
- [CLI](#)

```
New-AzResourceGroupDeployment -ResourceGroupName "<resource-group-name>" -TemplateFile "<template-filename.json>" -TemplateParameterFile "<parameter-filename.json>"
```

# Use PowerShell

You can install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers by using the PowerShell command for adding a virtual machine extension.

## Install on Azure virtual machines

Use the following PowerShell commands to install the Azure Monitor agent on Azure virtual machines. Choose the appropriate command based on your chosen authentication method.

### User-assigned managed identity

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Location <location> -TypeHandlerVersion <version-number> -SettingString '{"authentication":{"managedIdentity":{"identifier-name":"mi_res_id","identifier-value":/subscriptions/<my-subscription-id>/resourceGroups/<my-resource-group>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<my-user-assigned-identity>"}}}'
```

### System-assigned managed identity

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Location <location> -TypeHandlerVersion <version-number>
```

### Uninstall on Azure virtual machines

Use the following PowerShell commands to uninstall the Azure Monitor agent on Azure virtual machines.

- [Windows](#)
- [Linux](#)

```
Remove-AzVMExtension -Name AzureMonitorWindowsAgent -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name>
```

### Update on Azure virtual machines

To perform a one-time update of the agent, you must first uninstall the existing agent version,. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -ExtensionName AzureMonitorWindowsAgent -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Publisher Microsoft.Azure.Monitor -ExtensionType AzureMonitorWindowsAgent -TypeHandlerVersion <version-number> -Location <location> -EnableAutomaticUpgrade $true
```

### Install on Azure Arc-enabled servers

Use the following PowerShell commands to install the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
New-AzConnectedMachineExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -MachineName <arc-server-name> -Location <arc-server-location>
```

### Uninstall on Azure Arc-enabled servers

Use the following PowerShell commands to uninstall the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
Remove-AzConnectedMachineExtension -MachineName <arc-server-name> -ResourceGroupName <resource-group-name> -Name AzureMonitorWindowsAgent
```

## Upgrade on Azure Arc-enabled servers

To perform a one-time upgrade of the agent, use the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
$target = @{"Microsoft.Azure.Monitor.AzureMonitorWindowsAgent" = @{"targetVersion"=<target-version-number>}}
Update-AzConnectedExtension -ResourceGroupName $env.ResourceGroupName -MachineName <arc-server-name> -ExtensionTarget $target
```

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade \(preview\)](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
Update-AzConnectedMachineExtension -ResourceGroup <resource-group-name> -MachineName <arc-server-name> -Name AzureMonitorWindowsAgent -EnableAutomaticUpgrade
```

# Use the Azure CLI

You can install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers by using the Azure CLI command for adding a virtual machine extension.

## Install on Azure virtual machines

Use the following CLI commands to install the Azure Monitor agent on Azure virtual machines. Choose the appropriate command based on your chosen authentication method.

### User-assigned managed identity

- [Windows](#)
- [Linux](#)

```
az vm extension set --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --ids <vm-resource-id> --settings '{"authentication":{"managedIdentity":{"identifier-name":"mi_res_id","identifier-value":/subscriptions/<my-subscription-id>/resourceGroups/<my-resource-group>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<my-user-assigned-identity>"}}}'
```

### System-assigned managed identity

- [Windows](#)
- [Linux](#)

```
az vm extension set --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --ids <vm-resource-id>
```

## Uninstall on Azure virtual machines

Use the following CLI commands to uninstall the Azure Monitor agent on Azure virtual machines.

- [Windows](#)
- [Linux](#)

```
az vm extension delete --resource-group <resource-group-name> --vm-name <virtual-machine-name> -name AzureMonitorWindowsAgent
```

## Update on Azure virtual machines

To perform a one-time update of the agent, you must first uninstall the existing agent version,. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature by using the following CLI commands.

- [Windows](#)
- [Linux](#)

```
az vm extension set -name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --vm-name <virtual-machine-name> --resource-group <resource-group-name> --enable-auto-upgrade true
```

## Install on Azure Arc-enabled servers

Use the following CLI commands to install the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension create --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --type AzureMonitorWindowsAgent --machine-name <arc-server-name> --resource-group <resource-group-name> --location <arc-server-location>
```

## Uninstall on Azure Arc-enabled servers

Use the following CLI commands to uninstall the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension delete --name AzureMonitorWindowsAgent --machine-name <arc-server-name> --resource-group <resource-group-name>
```

## Upgrade on Azure Arc-enabled servers

To perform a one-time upgrade of the agent, use the following CLI commands.

- [Windows](#)
- [Linux](#)

```
az connectedmachine upgrade-extension --extension-targets "  
{\"Microsoft.Azure.Monitor.AzureMonitorWindowsAgent\":{\"targetVersion\":\"<target-version-number>\"},\"}  
--machine-name <arc-server-name> --resource-group <resource-group-name>
```

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade \(preview\)](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension update --name AzureMonitorWindowsAgent --machine-name <arc-server-name> --  
resource-group <resource-group-name> --enable-auto-upgrade true
```

## Use Azure Policy

Use the following policies and policy initiatives to automatically install the agent and associate it with a data collection rule every time you create a virtual machine, scale set, or Azure Arc-enabled server.

### NOTE

As per Microsoft Identity best practices, policies for installing the Azure Monitor agent on virtual machines and scale sets rely on user-assigned managed identity. This option is the more scalable and resilient managed identity for these resources. For Azure Arc-enabled servers, policies rely on system-assigned managed identity as the only supported option today.

### Built-in policy initiatives

Before you proceed, review [prerequisites for agent installation](#).

Policy initiatives for Windows and Linux virtual machines, scale sets consist of individual policies that:

- (Optional) Create and assign built-in user-assigned managed identity, per subscription, per region. [Learn more](#).
  - **Bring Your Own User-Assigned Identity** : If set to `true`, it creates the built-in user-assigned managed identity in the predefined resource group and assigns it to all machines that the policy is applied to. If set to `false`, you can instead use existing user-assigned identity that *you must assign* to the machines beforehand.
- Install the Azure Monitor agent extension on the machine, and configure it to use user-assigned identity as specified by the following parameters.
  - **Bring Your Own User-Assigned Managed Identity** : If set to `false`, it configures the agent to use the built-in user-assigned managed identity created by the preceding policy. If set to `true`, it configures the agent to use an existing user-assigned identity that *you must assign* to the machines in scope beforehand.
  - **User-Assigned Managed Identity Name** : If you use your own identity (selected `true`), specify the name of the identity that's assigned to the machines.
  - **User-Assigned Managed Identity Resource Group** : If you use your own identity (selected `true`), specify the resource group where the identity exists.
  - **Additional Virtual Machine Images** : Pass additional VM image names that you want to apply the policy to, if not already included.
- Create and deploy the association to link the machine to specified data collection rule.

- **Data Collection Rule Resource Id** : The Azure Resource Manager resourceId of the rule you want to associate via this policy to all machines the policy is applied to.

Home > Policy >

**Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection Rule**

Initiative Definition

[Assign](#) [Edit initiative](#) [Duplicate initiative](#) [Delete initiative](#) [Export initiative](#)

[Essentials](#)

Name	: Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection ...	Definition location	: --
Description	: Monitor your Windows virtual machines and virtual machine scale sets by deploying the Azure Monitor Agent extension wit...	Definition ID	: /providers/Microsoft.Authorization/policySetDefinitions/0d1b56c6-6d11-4a5d-8
Category	: Monitoring	Type	: Built-in
Version	: 1.0.0		

[Automated](#) Microsoft managed Attestation Assignments (0) Parameters

Filter by reference ID, policy name... All effects All types

Policy	Effect Type	Type	Reference ID
[Assign Built-In User-Assigned Managed Identity to Virtual Machines]	[parameters('effect')]	Built-in	addUserAssignedManagedIdentity
[Assign Built-In User-Assigned Managed Identity to Virtual Machine Scale Sets]	[parameters('effect')]	Built-in	addUserAssignedManagedIdentity
[Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication]	[parameters('effect')]	Built-in	deployAzureMonitoringAgent
[Configure Windows virtual machine scale sets to run Azure Monitor Agent with user-assigned managed identity-based authentication]	[parameters('effect')]	Built-in	deployAzureMonitoringAgentWi
[Configure Windows Machines to be associated with a Data Collection Rule]	[parameters('effect')]	Built-in	associateDataCollectionRuleV

## Known issues

- Managed Identity default behavior. [Learn more](#).
- Possible race condition with using built-in user-assigned identity creation policy. [Learn more](#).
- Assigning policy to resource groups. If the assignment scope of the policy is a resource group and not a subscription, the identity used by policy assignment (different from the user-assigned identity used by agent) must be manually granted [these roles](#) prior to assignment/remediation. Failing to do this step will result in *deployment failures*.
- Other [Managed Identity limitations](#).

## Built-in policies

You can choose to use the individual policies from the preceding policy initiative to perform a single action at scale. For example, if you *only* want to automatically install the agent, use the second agent installation policy from the initiative, as shown.

Home > Policy > Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection Rule >

**Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication**

Policy definition

[Assign](#) [Edit definition](#) [Duplicate definition](#) [Delete definition](#) [Export definition](#)

[Essentials](#)

Name	: Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based auth...	Definition location	: --
Description	: Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telem...	Definition ID	: /providers/Microsoft.Authorization/policyDefinitions/637125fd-7c39-4b
Available Effects	: DeployIfNotExists, Disabled	Type	: Built-in
Category	: Monitoring	Mode	: Indexed

[Definition](#) Assignments (0) Parameters

```

1 {
2   "properties": {
3     "displayName": "Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication",
4     "policyType": "BuiltIn",
5     "mode": "Indexed",
6     "description": "Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telemetry data from the guest OS. This policy is triggered when a new VM is created or updated. It deploys the Azure Monitor Agent extension to the VM and associates it with a Data Collection Rule defined in the same policy set. This ensures that the agent is installed and configured automatically for every new VM instance."}
```

## Remediation

The initiatives or policies will apply to each virtual machine as it's created. A [remediation task](#) deploys the policy definitions in the initiative to existing resources, so you can configure the Azure Monitor agent for any resources that were already created.

When you create the assignment by using the Azure portal, you have the option of creating a remediation task at the same time. For information on the remediation, see [Remediate non-compliant resources with Azure Policy](#).

## Configure Azure Monitor Agent to Linux virtual machines and associate to Data Collection Rule

[Assign initiative](#)[Basics](#)   [Parameters](#)   [Remediation](#)   [Non-compliance messages](#)   [Review + create](#)

By default, this assignment will only take effect on newly created resources. Existing resources can be updated via a remediation task after the policy is assigned. For deployIfNotExists policies, the remediation task will deploy the specified template. For modify policies, the remediation task will edit tags on the existing resources.

 Create a remediation task [\(?\)](#)

Policy to remediate

[Configure Linux virtual machines with Azure Monitor Agent](#) [\(?\)](#)

### Managed Identity

Policies with the deployIfNotExists and modify effect types need the ability to deploy resources and edit tags on existing resources respectively. To do this, a managed identity will be created for you.

[Learn more about Managed Identity.](#) Create a Managed Identity [\(?\)](#)

Managed identity location \*

[East US](#) [\(?\)](#)

### Permissions

This identity will also be given the following permissions:

[Virtual Machine Contributor, Monitoring Contributor](#) [\(?\)](#)

 Role assignments (permissions) are created based on the role definitions specified in the policies.

[Review + create](#)[Cancel](#)[Previous](#)[Next](#)

## Next steps

[Create a data collection rule](#) to collect data from the agent and send it to Azure Monitor.

# Azure Diagnostics extension overview

9/21/2022 • 5 minutes to read • [Edit Online](#)

Azure Diagnostics extension is an [agent in Azure Monitor](#) that collects monitoring data from the guest operating system of Azure compute resources including virtual machines. This article provides an overview of Azure Diagnostics extension including specific functionality that it supports and options for installation and configuration.

## NOTE

Azure Diagnostics extension is one of the agents available to collect monitoring data from the guest operating system of compute resources. See [Overview of the Azure Monitor agents](#) for a description of the different agents and guidance on selecting the appropriate agents for your requirements.

## Primary scenarios

The primary scenarios addressed by the diagnostics extension are:

Use the Azure Diagnostics extension if you need to:

- Send data to Azure Storage for archiving or to analyze it with tools such as [Azure Storage Explorer](#).
- Send data to [Azure Monitor Metrics](#) to analyze it with [Metrics Explorer](#) and to take advantage of features such as near-real-time [metric alerts](#) and [autoscale](#) (Windows only).
- Send data to third-party tools by using [Azure Event Hubs](#).
- Collect [Boot Diagnostics](#) to investigate VM boot issues.

Limitations of the Azure Diagnostics extension:

- Can only be used with Azure resources.
- Limited ability to send data to Azure Monitor Logs.

## Comparison to Log Analytics agent

The Log Analytics agent in Azure Monitor can also be used to collect monitoring data from the guest operating system of virtual machines. You may choose to use either or both depending on your requirements. See [Overview of the Azure Monitor agents](#) for a detailed comparison of the Azure Monitor agents.

The key differences to consider are:

- Azure Diagnostics Extension can be used only with Azure virtual machines. The Log Analytics agent can be used with virtual machines in Azure, other clouds, and on-premises.
- Azure Diagnostics extension sends data to Azure Storage, [Azure Monitor Metrics](#) (Windows only) and Event Hubs. The Log Analytics agent collects data to [Azure Monitor Logs](#).
- The Log Analytics agent is required for [solutions](#), [VM insights](#), and other services such as [Microsoft Defender for Cloud](#).

## Costs

There is no cost for Azure Diagnostic Extension, but you may incur charges for the data ingested. Check [Azure Monitor pricing](#) for the destination where you're collecting data.

## Data collected

The following tables list the data that can be collected by the Windows and Linux diagnostics extension.

### Windows diagnostics extension (WAD)

DATA SOURCE	DESCRIPTION
Windows Event logs	Events from Windows event log.
Performance counters	Numerical values measuring performance of different aspects of operating system and workloads.
IIS Logs	Usage information for IIS web sites running on the guest operating system.
Application logs	Trace messages written by your application.
.NET EventSource logs	Code writing events using the .NET <a href="#">EventSource</a> class
Manifest based ETW logs	Event Tracing for Windows events generated by any process.
Crash dumps (logs)	Information about the state of the process if an application crashes.
File based logs	Logs created by your application or service.
Agent diagnostic logs	Information about Azure Diagnostics itself.

### Linux diagnostics extension (LAD)

DATA SOURCE	DESCRIPTION
Syslog	Events sent to the Linux event logging system.
Performance counters	Numerical values measuring performance of different aspects of operating system and workloads.
Log files	Entries sent to a file based log.

## Data destinations

The Azure Diagnostic extension for both Windows and Linux always collect data into an Azure Storage account.

See [Install and configure Windows Azure diagnostics extension \(WAD\)](#) and [Use Linux Diagnostic Extension to monitor metrics and logs](#) for a list of specific tables and blobs where this data is collected.

Configure one or more *data sinks* to send data to other additional destinations. The following sections list the sinks available for the Windows and Linux diagnostics extension.

### Windows diagnostics extension (WAD)

DESTINATION	DESCRIPTION

DESTINATION	DESCRIPTION
Azure Monitor Metrics	Collect performance data to Azure Monitor Metrics. See <a href="#">Send Guest OS metrics to the Azure Monitor metric database</a> .
Event hubs	Use Azure Event Hubs to send data outside of Azure. See <a href="#">Streaming Azure Diagnostics data to Event Hubs</a>
Azure Storage blobs	Write to data to blobs in Azure Storage in addition to tables.
Application Insights	Collect data from applications running in your VM to Application Insights to integrate with other application monitoring. See <a href="#">Send diagnostic data to Application Insights</a> .

You can also collect WAD data from storage into a Log Analytics workspace to analyze it with Azure Monitor Logs although the Log Analytics agent is typically used for this functionality. It can send data directly to a Log Analytics workspace and supports solutions and insights that provide additional functionality. See [Collect Azure diagnostic logs from Azure Storage](#).

### Linux diagnostics extension (LAD)

LAD writes data to tables in Azure Storage. It supports the sinks in the following table.

DESTINATION	DESCRIPTION
Event hubs	Use Azure Event Hubs to send data outside of Azure.
Azure Storage blobs	Write to data to blobs in Azure Storage in addition to tables.
Azure Monitor Metrics	Install the Telegraf agent in addition to LAD. See <a href="#">Collect custom metrics for a Linux VM with the InfluxData Telegraf agent</a> .

## Installation and configuration

The Diagnostic extension is implemented as a [virtual machine extension](#) in Azure, so it supports the same installation options using Resource Manager templates, PowerShell, and CLI. See [Virtual machine extensions and features for Windows](#) and [Virtual machine extensions and features for Linux](#) for general details on installing and maintaining virtual machine extensions.

You can also install and configure both the Windows and Linux diagnostic extension in the Azure portal under **Diagnostic settings** in the **Monitoring** section of the virtual machine's menu.

See the following articles for details on installing and configuring the diagnostics extension for Windows and Linux.

- [Install and configure Windows Azure diagnostics extension \(WAD\)](#)
- [Use Linux Diagnostic Extension to monitor metrics and logs](#)

## Other documentation

### Azure Cloud Service (classic) Web and Worker Roles

- [Introduction to Cloud Service Monitoring](#)
- [Enabling Azure Diagnostics in Azure Cloud Services](#)

- Application Insights for Azure cloud services  
[Trace the flow of a Cloud Services application with Azure Diagnostics](#)

## Azure Service Fabric

- Monitor and diagnose services in a local machine development setup

## Next steps

- Learn to [use Performance Counters in Azure Diagnostics](#).
- If you have trouble with diagnostics starting or finding your data in Azure storage tables, see [TroubleShooting Azure Diagnostics](#)

# Log Analytics virtual machine extension for Linux

9/21/2022 • 7 minutes to read • [Edit Online](#)

## Overview

Azure Monitor Logs provides monitoring, alerting, and alert remediation capabilities across cloud and on-premises assets. The Log Analytics virtual machine extension for Linux is published and supported by Microsoft. The extension installs the Log Analytics agent on Azure virtual machines, and enrolls virtual machines into an existing Log Analytics workspace. This document details the supported platforms, configurations, and deployment options for the Log Analytics virtual machine extension for Linux.

### NOTE

Azure Arc-enabled servers enables you to deploy, remove, and update the Log Analytics agent VM extension to non-Azure Windows and Linux machines, simplifying the management of your hybrid machine through their lifecycle. For more information, see [VM extension management with Azure Arc-enabled servers](#).

## Prerequisites

### Operating system

For details about the supported Linux distributions, refer to the [Overview of Azure Monitor agents](#) article.

### Agent and VM Extension version

The following table provides a mapping of the version of the Log Analytics VM extension and Log Analytics agent bundle for each release. A link to the release notes for the Log Analytics agent bundle version is included. Release notes include details on bug fixes and new features available for a given agent release.

LOG ANALYTICS LINUX VM EXTENSION VERSION	LOG ANALYTICS AGENT BUNDLE VERSION
1.14.19	<a href="#">1.14.19</a>
1.14.16	<a href="#">1.14.16</a>
1.14.13	<a href="#">1.14.13</a>
1.14.11	<a href="#">1.14.11</a>
1.14.9	<a href="#">1.14.9</a>
1.13.40	<a href="#">1.13.40</a>
1.13.35	<a href="#">1.13.35</a>
1.13.33	<a href="#">1.13.33</a>
1.13.27	<a href="#">1.13.27</a>
1.13.15	<a href="#">1.13.9-0</a>

LOG ANALYTICS LINUX VM EXTENSION VERSION	LOG ANALYTICS AGENT BUNDLE VERSION
1.12.25	<a href="#">1.12.15-0</a>
1.11.15	<a href="#">1.11.0-9</a>
1.10.0	<a href="#">1.10.0-1</a>
1.9.1	<a href="#">1.9.0-0</a>
1.8.11	<a href="#">1.8.1-256</a>
1.8.0	<a href="#">1.8.0-256</a>
1.7.9	<a href="#">1.6.1-3</a>
1.6.42.0	<a href="#">1.6.0-42</a>
1.4.60.2	<a href="#">1.4.4-210</a>
1.4.59.1	<a href="#">1.4.3-174</a>
1.4.58.7	<a href="#">14.2-125</a>
1.4.56.5	<a href="#">1.4.2-124</a>
1.4.55.4	<a href="#">1.4.1-123</a>
1.4.45.3	<a href="#">1.4.1-45</a>
1.4.45.2	<a href="#">1.4.0-45</a>
1.3.127.5	<a href="#">1.3.5-127</a>
1.3.127.7	<a href="#">1.3.5-127</a>
1.3.18.7	<a href="#">1.3.4-15</a>

## Microsoft Defender for Cloud

Microsoft Defender for Cloud automatically provisions the Log Analytics agent and connects it to a default Log Analytics workspace created by Defender for Cloud in your Azure subscription. If you are using Microsoft Defender for Cloud, do not run through the steps in this document. Doing so overwrites the configured workspace and breaks the connection with Microsoft Defender for Cloud.

## Internet connectivity

The Log Analytics agent extension for Linux requires that the target virtual machine is connected to the internet.

## Extension schema

The following JSON shows the schema for the Log Analytics agent extension. The extension requires the workspace ID and workspace key from the target Log Analytics workspace; these values can be [found in your Log Analytics workspace](#) in the Azure portal. Because the workspace key should be treated as sensitive data, it should be stored in a protected setting configuration. Azure VM extension protected setting data is encrypted,

and only decrypted on the target virtual machine. Note that **workspaceId** and **workspaceKey** are case-sensitive.

#### NOTE

Because the [Container Monitoring solution](#) is being retired, the following documentation uses the optional setting "skipDockerProviderInstall": true.

```
{  
  "type": "Microsoft.Compute/virtualMachines/extensions",  
  "name": "OMSExtension",  
  "apiVersion": "2018-06-01",  
  "location": "<location>",  
  "dependsOn": [  
    "[concat('Microsoft.Compute/virtualMachines/', <vm-name>)]"  
  ],  
  "properties": {  
    "publisher": "Microsoft.EnterpriseCloud.Monitoring",  
    "type": "OmsAgentForLinux",  
    "typeHandlerVersion": "1.13",  
    "autoUpgradeMinorVersion": true,  
    "settings": {  
      "workspaceId": "myWorkspaceId",  
      "skipDockerProviderInstall": true  
    },  
    "protectedSettings": {  
      "workspaceKey": "myWorkSpaceKey"  
    }  
  }  
}
```

#### NOTE

The schema above assumes that it will be placed at the root level of the template. If you put it inside the virtual machine resource in the template, the `type` and `name` properties should be changed, as described [further down](#).

## Property values

NAME	VALUE / EXAMPLE
apiVersion	2018-06-01
publisher	Microsoft.EnterpriseCloud.Monitoring
type	OmsAgentForLinux
typeHandlerVersion	1.13
workspaceId (e.g.)	6f680a37-00c6-41c7-a93f-1437e3462574
workspaceKey (e.g.)	z4bU3p1/GrnWpQkky4gdabWXAhbWSTz70hm4m2Xt92XI+rSRgE8qVvRhsGo9TXffbrTahyrvw35W0pOqQAU7uQ==

## Template deployment

#### NOTE

Certain components of the Log Analytics VM extension are also shipped in the [Diagnostics VM extension](#). Due to this architecture, conflicts can arise if both extensions are instantiated in the same ARM template. To avoid these install-time conflicts, use the `dependsOn` directive to ensure the extensions are installed sequentially. The extensions can be installed in either order.

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment configuration such as onboarding to Azure Monitor Logs. A sample Resource Manager template that includes the Log Analytics agent VM extension can be found on the [Azure Quickstart Gallery](#).

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the VM extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "type": "extensions",
  "name": "OMSExtension",
  "apiVersion": "2018-06-01",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <vm-name>)]"
  ],
  "properties": {
    "publisher": "Microsoft.EnterpriseCloud.Monitoring",
    "type": "OmsAgentForLinux",
    "typeHandlerVersion": "1.13",
    "settings": {
      "workspaceId": "myWorkspaceId",
      "skipDockerProviderInstall": true
    },
    "protectedSettings": {
      "workspaceKey": "myWorkSpaceKey"
    }
  }
}
```

When placing the extension JSON at the root of the template, the resource name includes a reference to the parent virtual machine, and the type reflects the nested configuration.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "<parentVmResource>/OMSExtension",
  "apiVersion": "2018-06-01",
  "location": "<location>",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', <vm-name>)]"
  ],
  "properties": {
    "publisher": "Microsoft.EnterpriseCloud.Monitoring",
    "type": "OmsAgentForLinux",
    "typeHandlerVersion": "1.13",
    "settings": {
      "workspaceId": "myWorkspaceId",
      "skipDockerProviderInstall": true
    },
    "protectedSettings": {
      "workspaceKey": "myWorkSpaceKey"
    }
  }
}
```

## Azure CLI deployment

The Azure CLI can be used to deploy the Log Analytics agent VM extension to an existing virtual machine. Replace the *myWorkspaceKey* value below with your workspace key and the *myWorkspaceId* value with your workspace ID. These values can be found in your Log Analytics workspace in the Azure portal under *Advanced Settings*. Replace the *latestVersion* value with a version from [Log Analytics Linux VM extension version](#).

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name OmsAgentForLinux \
--publisher Microsoft.EnterpriseCloud.Monitoring \
--protected-settings '{"workspaceKey":"myWorkSpaceKey"}' \
--settings '{"workspaceId":"myWorkspaceId","skipDockerProviderInstall": true}' \
--version latestVersion
```

## Azure PowerShell deployment

The Azure Powershell cmdlets can be used to deploy the Log Analytics agent VM extension to an existing virtual machine. Replace the *myWorkspaceKey* value below with your workspace key and the *myWorkspaceId* value with your workspace ID. These values can be found in your Log Analytics workspace in the Azure portal under *Advanced Settings*. Replace the *latestVersion* value with a version from [Log Analytics Linux VM extension version](#).

```
Set-AzVMExtension \
-ResourceGroupName myResourceGroup \
-VMName myVM \
-ExtensionName OmsAgentForLinux \
-ExtensionType OmsAgentForLinux \
-Publisher Microsoft.EnterpriseCloud.Monitoring \
-TypeHandlerVersion latestVersion \
-ProtectedSettingString '{"workspaceKey":"myWorkSpaceKey"}' \
-SettingString '{"workspaceId":"myWorkspaceId","skipDockerProviderInstall": true}'
```

## Troubleshoot and support

## Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI or Azure Powershell. To see the deployment state of extensions for a given VM, run the following command if you are using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file:

```
/var/log/azure/Microsoft.EnterpriseCloud.Monitoring.OmsAgentForLinux/extension.log
```

To retrieve the OMS extension version installed on a VM, run the following command if you are using Azure CLI.

```
az vm extension show --resource-group myResourceGroup --vm-name myVM -instance-view
```

To retrieve the OMS extension version installed on a VM, run the following command if you are using Azure PowerShell.

```
Get-AzVMExtension -ResourceGroupName my_resource_group -VMName my_vm_name -Name OmsAgentForLinux -Status
```

### Error codes and their meanings

ERROR CODE	MEANING	POSSIBLE ACTION
9	Enable called prematurely	<a href="#">Update the Azure Linux Agent</a> to the latest available version.
10	VM is already connected to a Log Analytics workspace	To connect the VM to the workspace specified in the extension schema, set stopOnMultipleConnections to false in public settings or remove this property. This VM gets billed once for each workspace it is connected to.
11	Invalid config provided to the extension	Follow the preceding examples to set all property values necessary for deployment.
17	Log Analytics package installation failure	
18	Installation of OMSConfig package failed.	Look through the command output for the root failure.
19	OMI package installation failure	
20	SCX package installation failure	
33	Error generating metaconfiguration for omsconfig.	File a <a href="#">GitHub Issue</a> with details from the output.
51	This extension is not supported on the VM's operation system	

ERROR CODE	MEANING	POSSIBLE ACTION
52	This extension failed due to a missing dependency or permission	Check the output and logs for more information about which dependency or permission is missing.
53	This extension failed due to missing or wrong configuration parameters	Check the output and logs for more information about what went wrong. Additionally, check the correctness of the workspace ID, and verify that the machine is connected to the internet.
55	Cannot connect to the Azure Monitor service or required packages missing or dpkg package manager is locked	Check that the system either has internet access, or that a valid HTTP proxy has been provided. Additionally, check the correctness of the workspace ID, and verify that curl and tar utilities are installed.

Additional troubleshooting information can be found on the [Log Analytics-Agent-for-Linux Troubleshooting Guide](#).

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Log Analytics virtual machine extension for Windows

9/21/2022 • 5 minutes to read • [Edit Online](#)

Azure Monitor Logs provides monitoring capabilities across cloud and on-premises assets. The Log Analytics agent virtual machine extension for Windows is published and supported by Microsoft. The extension installs the Log Analytics agent on Azure virtual machines, and enrolls virtual machines into an existing Log Analytics workspace. This document details the supported platforms, configurations, and deployment options for the Log Analytics virtual machine extension for Windows.

## NOTE

Azure Arc-enabled servers enables you to deploy, remove, and update the Log Analytics agent VM extension to non-Azure Windows and Linux machines, simplifying the management of your hybrid machine through their lifecycle. For more information, see [VM extension management with Azure Arc-enabled servers](#).

## Prerequisites

### Operating system

For details about the supported Windows operating systems, refer to the [Overview of Azure Monitor agents](#) article.

### Agent and VM Extension version

The following table provides a mapping of the version of the Windows Log Analytics VM extension and Log Analytics agent for each release.

LOG ANALYTICS WINDOWS AGENT VERSION	LOG ANALYTICS WINDOWS VM EXTENSION VERSION	RELEASE DATE	RELEASE NOTES
10.20.18067.0	1.0.18067	March 2022	<ul style="list-style-type: none"><li>Bug fix for perf counters</li><li>Enhancements to Agent Troubleshooter</li></ul>
10.20.18064.0	1.0.18064	December 2021	<ul style="list-style-type: none"><li>Bug fix for intermittent crashes</li></ul>
10.20.18062.0	1.0.18062	November 2021	<ul style="list-style-type: none"><li>Minor bug fixes and stabilization improvements</li></ul>

LOG ANALYTICS WINDOWS AGENT VERSION	LOG ANALYTICS WINDOWS VM EXTENSION VERSION	RELEASE DATE	RELEASE NOTES
10.20.18053	1.0.18053.0	October 2020	<ul style="list-style-type: none"> <li>• New Agent Troubleshooter</li> <li>• Updates to how the agent handles certificate changes to Azure services</li> </ul>
10.20.18040	1.0.18040.2	August 2020	<ul style="list-style-type: none"> <li>• Resolves an issue on Azure Arc</li> </ul>
10.20.18038	1.0.18038	April 2020	<ul style="list-style-type: none"> <li>• Enables connectivity over Private Link using Azure Monitor Private Link Scopes</li> <li>• Adds ingestion throttling to avoid a sudden, accidental influx in ingestion to a workspace</li> <li>• Adds support for additional Azure Government clouds and regions</li> <li>• Resolves a bug where HealthService.exe crashed</li> </ul>
10.20.18029	1.0.18029	March 2020	<ul style="list-style-type: none"> <li>• Adds SHA-2 code signing support</li> <li>• Improves VM extension installation and management</li> <li>• Resolves a bug with Azure Arc-enabled servers integration</li> <li>• Adds a built-in troubleshooting tool for customer support</li> <li>• Adds support for additional Azure Government regions</li> </ul>
10.20.18018	1.0.18018	October 2019	<ul style="list-style-type: none"> <li>• Minor bug fixes and stabilization improvements</li> </ul>

LOG ANALYTICS WINDOWS AGENT VERSION	LOG ANALYTICS WINDOWS VM EXTENSION VERSION	RELEASE DATE	RELEASE NOTES
10.20.18011	1.0.18011	July 2019	<ul style="list-style-type: none"> <li>Minor bug fixes and stabilization improvements</li> <li>Increased MaxExpressionDepth to 10000</li> </ul>
10.20.18001	1.0.18001	June 2019	<ul style="list-style-type: none"> <li>Minor bug fixes and stabilization improvements</li> <li>Added ability to disable default credentials when making proxy connection (support for WINHTTP_AUTOLOGON_SECURITY_LEVEL_HIGH)</li> </ul>
10.19.13515	1.0.13515	March 2019	<ul style="list-style-type: none"> <li>Minor stabilization fixes</li> </ul>
10.19.10006	n/a	Dec 2018	<ul style="list-style-type: none"> <li>Minor stabilization fixes</li> </ul>
8.0.11136	n/a	Sept 2018	<ul style="list-style-type: none"> <li>Added support for detecting resource ID change on VM move</li> <li>Added Support for reporting resource ID when using non-extension install</li> </ul>
8.0.11103	n/a	April 2018	
8.0.11081	1.0.11081	Nov 2017	
8.0.11072	1.0.11072	Sept 2017	
8.0.11049	1.0.11049	Feb 2017	

## Microsoft Defender for Cloud

Microsoft Defender for Cloud automatically provisions the Log Analytics agent and connects it with the default Log Analytics workspace of the Azure subscription. If you are using Microsoft Defender for Cloud, do not run through the steps in this document. Doing so overwrites the configured workspace and break the connection with Microsoft Defender for Cloud.

## Internet connectivity

The Log Analytics agent extension for Windows requires that the target virtual machine is connected to the

internet.

## Extension schema

The following JSON shows the schema for the Log Analytics agent extension. The extension requires the workspace ID and workspace key from the target Log Analytics workspace. These can be found in the settings for the workspace in the Azure portal. Because the workspace key should be treated as sensitive data, it should be stored in a protected setting configuration. Azure VM extension protected setting data is encrypted, and only decrypted on the target virtual machine. Note that **workspaceld** and **workspaceKey** are case-sensitive.

```
{  
    "type": "extensions",  
    "name": "OMSExtension",  
    "apiVersion": "[variables('apiVersion')]",  
    "location": "[resourceGroup().location]",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"  
    ],  
    "properties": {  
        "publisher": "Microsoft.EnterpriseCloud.Monitoring",  
        "type": "MicrosoftMonitoringAgent",  
        "typeHandlerVersion": "1.0",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "workspaceId": "myWorkSpaceId"  
        },  
        "protectedSettings": {  
            "workspaceKey": "myworkspaceKey"  
        }  
    }  
}
```

### Property values

NAME	VALUE / EXAMPLE
apiVersion	2015-06-15
publisher	Microsoft.EnterpriseCloud.Monitoring
type	MicrosoftMonitoringAgent
typeHandlerVersion	1.0
workspaceld (e.g.)*	6f680a37-00c6-41c7-a93f-1437e3462574
workspaceKey (e.g.)	z4bU3p1/GrnWpQkky4gdabWXAhbWSTz70hm4m2Xt92XI+rSRgE8qVvRhsGo9TXffbrTahyrwv35W0pOqQAU7uQ==

\* The workspaceld is called the consumerId in the Log Analytics API.

#### NOTE

For additional properties see Azure [Connect Windows Computers to Azure Monitor](#).

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. The JSON schema detailed in the previous section can be used in an Azure Resource Manager template to run the Log Analytics agent extension during an Azure Resource Manager template deployment. A sample template that includes the Log Analytics agent VM extension can be found on the [Azure Quickstart Gallery](#).

**NOTE**

The template does not support specifying more than one workspace ID and workspace key when you want to configure the agent to report to multiple workspaces. To configure the agent to report to multiple workspaces, see [Add or remove a workspace](#).

The JSON for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the Log Analytics extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{  
    "type": "extensions",  
    "name": "OMSExtension",  
    "apiVersion": "[variables('apiVersion')]",  
    "location": "[resourceGroup().location]",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"  
    ],  
    "properties": {  
        "publisher": "Microsoft.EnterpriseCloud.Monitoring",  
        "type": "MicrosoftMonitoringAgent",  
        "typeHandlerVersion": "1.0",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "workspaceId": "myWorkSpaceId"  
        },  
        "protectedSettings": {  
            "workspaceKey": "myWorkspaceKey"  
        }  
    }  
}
```

When placing the extension JSON at the root of the template, the resource name includes a reference to the parent virtual machine, and the type reflects the nested configuration.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "<parentVmResource>/OMSExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.EnterpriseCloud.Monitoring",
    "type": "MicrosoftMonitoringAgent",
    "typeHandlerVersion": "1.0",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "workspaceId": "myWorkSpaceId"
    },
    "protectedSettings": {
      "workspaceKey": "myWorkspaceKey"
    }
  }
}
```

## PowerShell deployment

The `Set-AzVMExtension` command can be used to deploy the Log Analytics agent virtual machine extension to an existing virtual machine. Before running the command, the public and private configurations need to be stored in a PowerShell hash table.

```
$PublicSettings = @{"workspaceId" = "myWorkSpaceId"}
$ProtectedSettings = @{"workspaceKey" = "myWorkspaceKey"}

Set-AzVMExtension -ExtensionName "MicrosoftMonitoringAgent" ` 
  -ResourceGroupName "myResourceGroup" ` 
  -VMName "myVM" ` 
  -Publisher "Microsoft.EnterpriseCloud.Monitoring" ` 
  -ExtensionType "MicrosoftMonitoringAgent" ` 
  -TypeHandlerVersion 1.0 ` 
  -Settings $PublicSettings ` 
  -ProtectedSettings $ProtectedSettings ` 
  -Location WestUS
```

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure PowerShell module. To see the deployment state of extensions for a given VM, run the following command using the Azure PowerShell module.

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

Extension execution output is logged to files found in the following directory:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent\
```

### Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and](#)

[Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Azure Monitor Dependency virtual machine extension for Linux

9/21/2022 • 3 minutes to read • [Edit Online](#)

The Azure Monitor for VMs Map feature gets its data from the Microsoft Dependency agent. The Azure VM Dependency agent virtual machine extension for Linux is published and supported by Microsoft. The extension installs the Dependency agent on Azure virtual machines. This document details the supported platforms, configurations, and deployment options for the Azure VM Dependency agent virtual machine extension for Linux.

## Prerequisites

### Operating system

The Azure VM Dependency agent extension for Linux can be run against the supported operating systems listed in the [Supported operating systems](#) section of the Azure Monitor for VMs deployment article.

## Extension schema

The following JSON shows the schema for the Azure VM Dependency agent extension on an Azure Linux VM.

```
{
    "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "1.0.0.0",
    "parameters": {
        "vmName": {
            "type": "string",
            "metadata": {
                "description": "The name of existing Linux Azure VM."
            }
        }
    },
    "variables": {
        "vmExtensionsApiVersion": "2017-03-30"
    },
    "resources": [
        {
            "type": "Microsoft.Compute/virtualMachines/extensions",
            "name": "[concat(parameters('vmName'), '/DAExtension')]",
            "apiVersion": "[variables('vmExtensionsApiVersion')]",
            "location": "[resourceGroup().location]",
            "dependsOn": [
            ],
            "properties": {
                "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",
                "type": "DependencyAgentLinux",
                "typeHandlerVersion": "9.5",
                "autoUpgradeMinorVersion": true
            }
        }
    ],
    "outputs": {}
}
```

## Property values

NAME	VALUE/EXAMPLE
apiVersion	2015-01-01
publisher	Microsoft.Azure.Monitoring.DependencyAgent
type	DependencyAgentLinux
typeHandlerVersion	9.5

## Template deployment

You can deploy Azure VM extensions with Azure Resource Manager templates. You can use the JSON schema detailed in the previous section in an Azure Resource Manager template to run the Azure VM Dependency agent extension during an Azure Resource Manager template deployment.

The JSON for a virtual machine extension can be nested inside the virtual machine resource. Or, you can place it at the root or top level of a Resource Manager JSON template. The placement of the JSON affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the Dependency agent extension is nested inside the virtual machine resource. When you nest the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "type": "extensions",
  "name": "DAExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",
    "type": "DependencyAgentLinux",
    "typeHandlerVersion": "9.5",
    "autoUpgradeMinorVersion": true
  }
}
```

When you place the extension JSON at the root of the template, the resource name includes a reference to the parent virtual machine. The type reflects the nested configuration.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "<parentVmResource>/DAExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",
    "type": "DependencyAgentLinux",
    "typeHandlerVersion": "9.5",
    "autoUpgradeMinorVersion": true
  }
}
```

# Azure CLI deployment

You can use the Azure CLI to deploy the Dependency agent VM extension to an existing virtual machine.

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name DependencyAgentLinux \
--publisher Microsoft.Azure.Monitoring.DependencyAgent \
--version 9.5
```

## Automatic extension upgrade

A new feature to [automatically upgrade minor versions](#) of Dependency extension is now available.

To enable automatic extension upgrade for an extension, you must ensure the property `enableAutomaticUpgrade` is set to `true` and added to the extension template. This property must be enabled on every VM or VM scale set individually. Use one of the methods described in the [enablement](#) section enable the feature for your VM or VM scale set.

When automatic extension upgrade is enabled on a VM or VM scale set, the extension is upgraded automatically whenever the extension publisher releases a new version for that extension. The upgrade is applied safely following availability-first principles as described [here](#).

The `enableAutomaticUpgrade` attribute's functionality is different from that of the `autoUpgradeMinorVersion`. The `autoUpgradeMinorVersion` attributes does not automatically trigger a minor version update when the extension publisher releases a new version. The `autoUpgradeMinorVersion` attribute indicates whether the extension should use a newer minor version if one is available at deployment time. Once deployed, however, the extension will not upgrade minor versions unless redeployed, even with this property set to true.

To keep your extension version updated, we recommend using `enableAutomaticUpgrade` with your extension deployment.

### IMPORTANT

If you add the `enableAutomaticUpgrade` to your template, make sure that you use at API version 2019-12-01 or higher.

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command by using the Azure CLI:

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file:

```
/var/opt/microsoft/dependency-agent/log/install.log
```

### Support

If you need more help at any point in this article, contact the Azure experts on the [Microsoft Q & A and Stack](#)

[Overflow forums](#). Or, you can file an Azure support incident. Go to the [Azure support site](#) and select **Get support**. For information about how to use Azure Support, read the [Microsoft Azure support FAQ](#).

# Azure Monitor Dependency virtual machine extension for Windows

9/21/2022 • 4 minutes to read • [Edit Online](#)

The Azure Monitor for VMs Map feature gets its data from the Microsoft Dependency agent. The Azure VM Dependency agent virtual machine extension for Windows is published and supported by Microsoft. The extension installs the Dependency agent on Azure virtual machines. This document details the supported platforms, configurations, and deployment options for the Azure VM Dependency agent virtual machine extension for Windows.

## Operating system

The Azure VM Dependency agent extension for Windows can be run against the supported operating systems listed in the [Supported operating systems](#) section of the Azure Monitor for VMs deployment article.

## Extension schema

The following JSON shows the schema for the Azure VM Dependency agent extension on an Azure Windows VM.

```
{  
    "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",  
    "contentVersion": "1.0.0.0",  
    "parameters": {  
        "vmName": {  
            "type": "string",  
            "metadata": {  
                "description": "The name of existing Azure VM. Supported Windows Server versions: 2008 R2 and above  
(x64)."  
            }  
        }  
    },  
    "variables": {  
        "vmExtensionsApiVersion": "2017-03-30"  
    },  
    "resources": [  
        {  
            "type": "Microsoft.Compute/virtualMachines/extensions",  
            "name": "[concat(parameters('vmName'), '/DAExtension')]",  
            "apiVersion": "[variables('vmExtensionsApiVersion')]",  
            "location": "[resourceGroup().location]",  
            "dependsOn": [  
            ],  
            "properties": {  
                "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",  
                "type": "DependencyAgentWindows",  
                "typeHandlerVersion": "9.10",  
                "autoUpgradeMinorVersion": true  
            }  
        }  
    ],  
    "outputs": {}  
}
```

## Property values

NAME	VALUE/EXAMPLE
apiVersion	2015-01-01
publisher	Microsoft.Azure.Monitoring.DependencyAgent
type	DependencyAgentWindows
typeHandlerVersion	9.10

## Template deployment

You can deploy the Azure VM extensions with Azure Resource Manager templates. You can use the JSON schema detailed in the previous section in an Azure Resource Manager template to run the Azure VM Dependency agent extension during an Azure Resource Manager template deployment.

The JSON for a virtual machine extension can be nested inside the virtual machine resource. Or, you can place it at the root or top level of a Resource Manager JSON template. The placement of the JSON affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the Dependency agent extension is nested inside the virtual machine resource. When you nest the extension resource, the JSON is placed in the `"resources": []` object of the virtual machine.

```
{
  "type": "extensions",
  "name": "DAExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",
    "type": "DependencyAgentWindows",
    "typeHandlerVersion": "9.10",
    "autoUpgradeMinorVersion": true
  }
}
```

When you place the extension JSON at the root of the template, the resource name includes a reference to the parent virtual machine. The type reflects the nested configuration.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "<parentVmResource>/DAExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",
    "type": "DependencyAgentWindows",
    "typeHandlerVersion": "9.10",
    "autoUpgradeMinorVersion": true
  }
}
```

## PowerShell deployment

You can use the `Set-AzVMExtension` command to deploy the Dependency agent virtual machine extension to an existing virtual machine. Before you run the command, the public and private configurations need to be stored in a PowerShell hash table.

```
Set-AzVMExtension -ExtensionName "Microsoft.Azure.Monitoring.DependencyAgent" ` 
  -ResourceGroupName "myResourceGroup" ` 
  -VMName "myVM" ` 
  -Publisher "Microsoft.Azure.Monitoring.DependencyAgent" ` 
  -ExtensionType "DependencyAgentWindows" ` 
  -TypeHandlerVersion 9.10 ` 
  -Location WestUS
```

## Automatic extension upgrade

A new feature to [automatically upgrade minor versions](#) of Dependency extension is now available.

To enable automatic extension upgrade for an extension, you must ensure the property `enableAutomaticUpgrade` is set to `true` and added to the extension template. This property must be enabled on every VM or VM scale set individually. Use one of the methods described in the [enablement](#) section enable the feature for your VM or VM scale set.

When automatic extension upgrade is enabled on a VM or VM scale set, the extension is upgraded automatically whenever the extension publisher releases a new version for that extension. The upgrade is applied safely following availability-first principles as described [here](#).

The `enableAutomaticUpgrade` attribute's functionality is different from that of the `autoUpgradeMinorVersion`. The `autoUpgradeMinorVersion` attributes does not automatically trigger a minor version update when the extension publisher releases a new version. The `autoUpgradeMinorVersion` attribute indicates whether the extension should use a newer minor version if one is available at deployment time. Once deployed, however, the extension will not upgrade minor versions unless redeployed, even with this property set to true.

To keep your extension version updated, we recommend using `enableAutomaticUpgrade` with your extension deployment.

### IMPORTANT

If you add the `enableAutomaticUpgrade` to your template, make sure that you use at API version 2019-12-01 or higher.

# Troubleshoot and support

## Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal and by using the Azure PowerShell module. To see the deployment state of extensions for a given VM, run the following command by using the Azure PowerShell module:

```
Get-AzVMExtension -ResourceGroupName myResourceGroup -VMName myVM -Name myExtensionName
```

Extension execution output is logged to files found in the following directory:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.Azure.Monitoring.DependencyAgent\
```

## Support

If you need more help at any point in this article, you can contact the Azure experts on the [Microsoft Q & A and Stack Overflow forums](#). Or, you can file an Azure support incident. Go to the [Azure support site](#) and select **Get support**. For information about how to use Azure Support, read the [Microsoft Azure support FAQ](#).

# Introduction to the Azure Desired State Configuration extension handler

9/21/2022 • 9 minutes to read • [Edit Online](#)

The Azure VM Agent and associated extensions are part of Microsoft Azure infrastructure services. VM extensions are software components that extend VM functionality and simplify various VM management operations.

## NOTE

Before you enable the DSC extension, we would like you to know that a newer version of DSC is now available in preview, managed by a feature of Azure Policy named [guest configuration](#). The guest configuration feature combines features of the Desired State Configuration (DSC) extension handler, Azure Automation State Configuration, and the most commonly requested features from customer feedback. Guest configuration also includes hybrid machine support through [Arc-enabled servers](#).

The primary use case for the Azure Desired State Configuration (DSC) extension is to bootstrap a VM to the [Azure Automation State Configuration \(DSC\) service](#). The service provides [benefits](#) that include ongoing management of the VM configuration and integration with other operational tools, such as Azure Monitoring. Using the extension to register VM's to the service provides a flexible solution that even works across Azure subscriptions.

You can use the DSC extension independently of the Automation DSC service. However, this will only push a configuration to the VM. No ongoing reporting is available, other than locally in the VM.

This article provides information about both scenarios: using the DSC extension for Automation onboarding, and using the DSC extension as a tool for assigning configurations to VMs by using the Azure SDK.

## Prerequisites

- **Local machine:** To interact with the Azure VM extension, you must use either the Azure portal or the Azure PowerShell SDK.
- **Guest Agent:** The Azure VM that's configured by the DSC configuration must be an OS that supports Windows Management Framework (WMF) 4.0 or later. For the full list of supported OS versions, see the [DSC extension version history](#).

## Terms and concepts

This guide assumes familiarity with the following concepts:

- **Configuration:** A DSC configuration document.
- **Node:** A target for a DSC configuration. In this document, *node* always refers to an Azure VM.
- **Configuration data:** A .psd1 file that has environmental data for a configuration.

## Architecture

The Azure DSC extension uses the Azure VM Agent framework to deliver, enact, and report on DSC configurations running on Azure VMs. The DSC extension accepts a configuration document and a set of parameters. If no file is provided, a [default configuration script](#) is embedded with the extension. The default

configuration script is used only to set metadata in [Local Configuration Manager](#).

When the extension is called for the first time, it installs a version of WMF by using the following logic:

- If the Azure VM OS is Windows Server 2016, no action is taken. Windows Server 2016 already has the latest version of PowerShell installed.
- If the **wmfVersion** property is specified, that version of WMF is installed, unless that version is incompatible with the VM's OS.
- If no **wmfVersion** property is specified, the latest applicable version of WMF is installed.

Installing WMF requires a restart. After restarting, the extension downloads the .zip file that's specified in the **modulesUrl** property, if provided. If this location is in Azure Blob storage, you can specify an SAS token in the **sasToken** property to access the file. After the .zip is downloaded and unpacked, the configuration function defined in **configurationFunction** runs to generate an .mof([Managed Object Format](#)) file. The extension then runs `Start-DscConfiguration -Force` by using the generated .mof file. The extension captures output and writes it to the Azure status channel.

### Default configuration script

The Azure DSC extension includes a default configuration script that's intended to be used when you onboard a VM to the Azure Automation DSC service. The script parameters are aligned with the configurable properties of [Local Configuration Manager](#). For script parameters, see [Default configuration script in Desired State Configuration extension with Azure Resource Manager templates](#). For the full script, see the [Azure quickstart template in GitHub](#).

## Information for registering with Azure Automation State Configuration (DSC) service

When using the DSC Extension to register a node with the State Configuration service, three values will need to be provided.

- **RegistrationUrl** - the https address of the Azure Automation account
- **RegistrationKey** - a shared secret used to register nodes with the service
- **NodeConfigurationName** - the name of the Node Configuration (MOF) to pull from the service to configure the server role

This information can be seen in the Azure portal or you can use PowerShell.

```
(Get-AzAutomationRegistrationInfo -ResourceGroupName <resourcegroupname> -AutomationAccountName <accountname>).Endpoint  
(Get-AzAutomationRegistrationInfo -ResourceGroupName <resourcegroupname> -AutomationAccountName <accountname>).PrimaryKey
```

For the Node Configuration name, make sure the node configuration exists in Azure State Configuration. If it does not, the extension deployment will return a failure. Also make sure you are using the name of the *Node Configuration* and not the Configuration. A Configuration is defined in a script that is used [to compile the Node Configuration \(MOF file\)](#). The name will always be the Configuration followed by a period `.` and either `localhost` or a specific computer name.

## DSC extension in Resource Manager templates

In most scenarios, Resource Manager deployment templates are the expected way to work with the DSC extension. For more information and for examples of how to include the DSC extension in Resource Manager deployment templates, see [Desired State Configuration extension with Azure Resource Manager templates](#).

## DSC extension PowerShell cmdlets

The PowerShell cmdlets that are used to manage the DSC extension are best used in interactive troubleshooting and information-gathering scenarios. You can use the cmdlets to package, publish, and monitor DSC extension deployments. Cmdlets for the DSC extension aren't yet updated to work with the [default configuration script](#).

The **Publish-AzVMDscConfiguration** cmdlet takes in a configuration file, scans it for dependent DSC resources, and then creates a .zip file. The .zip file contains the configuration and DSC resources that are needed to enact the configuration. The cmdlet can also create the package locally by using the *-OutputArchivePath* parameter. Otherwise, the cmdlet publishes the .zip file to blob storage, and then secures it with an SAS token.

The .ps1 configuration script that the cmdlet creates is in the .zip file at the root of the archive folder. The module folder is placed in the archive folder in resources.

The **Set-AzVMDscExtension** cmdlet injects the settings that the PowerShell DSC extension requires into a VM configuration object.

The **Get-AzVMDscExtension** cmdlet retrieves the DSC extension status of a specific VM.

The **Get-AzVMDscExtensionStatus** cmdlet retrieves the status of the DSC configuration that's enacted by the DSC extension handler. This action can be performed on a single VM or on a group of VMs.

The **Remove-AzVMDscExtension** cmdlet removes the extension handler from a specific VM. This cmdlet does *not* remove the configuration, uninstall WMF, or change the applied settings on the VM. It only removes the extension handler.

Important information about Resource Manager DSC extension cmdlets:

- Azure Resource Manager cmdlets are synchronous.
- The *ResourceGroupName*, *VMName*, *ArchiveStorageAccountName*, *Version*, and *Location* parameters are all required.
- *ArchiveResourceGroupName* is an optional parameter. You can specify this parameter when your storage account belongs to a different resource group than the one where the VM is created.
- Use the **AutoUpdate** switch to automatically update the extension handler to the latest version when it's available. This parameter has the potential to cause restarts on the VM when a new version of WMF is released.

### Get started with cmdlets

The Azure DSC extension can use DSC configuration documents to directly configure Azure VMs during deployment. This step doesn't register the node to Automation. The node is *not* centrally managed.

The following example shows a simple example of a configuration. Save the configuration locally as `iisInstall.ps1`.

```
configuration IISInstall
{
    node "localhost"
    {
        WindowsFeature IIS
        {
            Ensure = "Present"
            Name = "Web-Server"
        }
    }
}
```

The following commands place the `iisInstall.ps1` script on the specified VM. The commands also execute the configuration, and then report back on status.

```

$resourceGroup = 'dscVmDemo'
$vmName = 'myVM'
$storageName = 'demostorage'
#Publish the configuration script to user storage
Publish-AzVMDscConfiguration -ConfigurationPath .\iisInstall.ps1 -ResourceGroupName $resourceGroup -
StorageAccountName $storageName -force
#Set the VM to run the DSC configuration
Set-AzVMDscExtension -Version '2.76' -ResourceGroupName $resourceGroup -VMName $vmName -
ArchiveStorageAccountName $storageName -ArchiveBlobName 'iisInstall.ps1.zip' -AutoUpdate -ConfigurationName
'IISInstall'

```

## Azure CLI deployment

The Azure CLI can be used to deploy the DSC extension to an existing virtual machine.

For a virtual machine running Windows:

```

az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name DSC \
--publisher Microsoft.PowerShell \
--version 2.77 --protected-settings '{}' \
--settings '{}'

```

For a virtual machine running Linux:

```

az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name DSCForLinux \
--publisher Microsoft.OSTCExtensions \
--version 2.7 --protected-settings '{}' \
--settings '{}'

```

## Azure portal functionality

To set up DSC in the portal:

1. Go to a VM.
2. Under **Settings**, select **Extensions**.
3. In the new page that's created, select + **Add**, and then select **PowerShell Desired State Configuration**.
4. Click **Create** at the bottom of the extension information page.

The portal collects the following input:

- **Configuration Modules or Script:** This field is mandatory (the form has not been updated for the [default configuration script](#)). Configuration modules and scripts require a .ps1 file that has a configuration script or a .zip file with a .ps1 configuration script at the root. If you use a .zip file, all dependent resources must be included in module folders in the .zip. You can create the .zip file by using the **Publish-AzureVMDscConfiguration -OutputArchivePath** cmdlet that's included in the Azure PowerShell SDK. The .zip file is uploaded to your user blob storage and secured by an SAS token.
- **Module-qualified Name of Configuration:** You can include multiple configuration functions in a .ps1 file. Enter the name of the configuration .ps1 script followed by \ and the name of the configuration function. For example, if your .ps1 script has the name configuration.ps1 and the configuration is

lisInstall, enter `configuration.ps1\lisInstall`.

- **Configuration Arguments:** If the configuration function takes arguments, enter them here in the format `argumentName1=value1,argumentName2=value2`. This format is a different format in which configuration arguments are accepted in PowerShell cmdlets or Resource Manager templates.
- **Configuration Data PSD1 File:** If your configuration requires a configuration data file in `.psd1`, use this field to select the data file and upload it to your user blob storage. The configuration data file is secured by an SAS token in blob storage.
- **WMF Version:** Specifies the version of Windows Management Framework (WMF) that should be installed on your VM. Setting this property to latest installs the most recent version of WMF. Currently, the only possible values for this property are 4.0, 5.0, 5.1, and latest. These possible values are subject to updates. The default value is **latest**.
- **Data Collection:** Determines if the extension will collect telemetry. For more information, see [Azure DSC extension data collection](#).
- **Version:** Specifies the version of the DSC extension to install. For information about versions, see [DSC extension version history](#).
- **Auto Upgrade Minor Version:** This field maps to the **AutoUpdate** switch in the cmdlets and enables the extension to automatically update to the latest version during installation. **Yes** will instruct the extension handler to use the latest available version and **No** will force the **Version** specified to be installed. Selecting neither **Yes** nor **No** is the same as selecting **No**.

## Logs

Logs for the extension are stored in the following location:

```
C:\WindowsAzure\Logs\Plugins\Microsoft.Powershell.DSC\<version number>
```

## Next steps

- For more information about PowerShell DSC, go to the [PowerShell documentation center](#).
- Examine the [Resource Manager template for the DSC extension](#).
- For more functionality that you can manage by using PowerShell DSC, and for more DSC resources, browse the [PowerShell gallery](#).
- For details about passing sensitive parameters into configurations, see [Manage credentials securely with the DSC extension handler](#).

# DSC extension for Linux (Microsoft.OSTCExtensions.DSCForLinux)

9/21/2022 • 7 minutes to read • [Edit Online](#)

Desired State Configuration (DSC) is a management platform that you can use to manage your IT and development infrastructure with configuration as code.

## NOTE

The DSC extension for Linux and the [Log Analytics virtual machine extension for Linux](#) currently present a conflict and aren't supported in a side-by-side configuration. Don't use the two solutions together on the same VM.

Before you enable the DSC extension, we would like you to know that a newer version of DSC is now available in preview, managed by a feature of Azure Policy named [guest configuration](#). The guest configuration feature combines features of the Desired State Configuration (DSC) extension handler, Azure Automation State Configuration, and the most commonly requested features from customer feedback. Guest configuration also includes hybrid machine support through [Arc-enabled servers](#).

The DSCForLinux extension is published and supported by Microsoft. The extension installs the OMI and DSC agent on Azure virtual machines. The DSC extension can also do the following actions:

- Register the Linux VM to an Azure Automation account to pull configurations from the Azure Automation service (Register ExtensionAction).
- Push MOF configurations to the Linux VM (Push ExtensionAction).
- Apply meta MOF configuration to the Linux VM to configure a pull server in order to pull node configuration (Pull ExtensionAction).
- Install custom DSC modules to the Linux VM (Install ExtensionAction).
- Remove custom DSC modules from the Linux VM (Remove ExtensionAction).

## Prerequisites

### Operating system

For nodes running Linux, the DSC Linux extension supports all the Linux distributions listed in the [PowerShell DSC documentation](#).

### Internet connectivity

The DSCForLinux extension requires the target virtual machine to be connected to the internet. For example, the Register extension requires connectivity to the Automation service. For other actions such as Pull, Pull, Install requires connectivity to Azure Storage and GitHub. It depends on settings provided by the customer.

## Extension schema

### Public configuration

Here are all the supported public configuration parameters:

- `FileUri` : (optional, string) The uri of the MOF file, meta MOF file, or custom resource zip file.
- `ResourceName` : (optional, string) The name of the custom resource module.
- `ExtensionAction` : (optional, string) Specifies what an extension does. Valid values are Register, Push, Pull, Install, and Remove. If not specified, it's considered a Push Action by default.

- `NodeConfigurationName` : (optional, string) The name of a node configuration to apply.
- `RefreshFrequencyMins` : (optional, int) Specifies how often (in minutes) that DSC attempts to obtain the configuration from the pull server. If configuration on the pull server differs from the current one on the target node, it's copied to the pending store and applied.
- `ConfigurationMode` : (optional, string) Specifies how DSC should apply the configuration. Valid values are `ApplyOnly`, `ApplyAndMonitor`, and `ApplyAndAutoCorrect`.
- `ConfigurationModeFrequencyMins` : (optional, int) Specifies how often (in minutes) DSC ensures that the configuration is in the desired state.

#### **NOTE**

If you use a version earlier than 2.3, the mode parameter is the same as ExtensionAction. Mode seems to be an overloaded term. To avoid confusion, ExtensionAction is used from version 2.3 onward. For backward compatibility, the extension supports both mode and ExtensionAction.

## Protected configuration

Here are all the supported protected configuration parameters:

- `StorageAccountName` : (optional, string) The name of the storage account that contains the file
- `StorageAccountKey` : (optional, string) The key of the storage account that contains the file
- `RegistrationUrl` : (optional, string) The URL of the Azure Automation account
- `RegistrationKey` : (optional, string) The access key of the Azure Automation account

## Scenarios

### Register an Azure Automation account

protected.json

```
{
  "RegistrationUrl": "<azure-automation-account-url>",
  "RegistrationKey": "<azure-automation-account-key>"
}
```

public.json

```
{
  "ExtensionAction" : "Register",
  "NodeConfigurationName" : "<node-configuration-name>",
  "RefreshFrequencyMins" : "<value>",
  "ConfigurationMode" : "<ApplyAndMonitor | ApplyAndAutoCorrect | ApplyOnly>",
  "ConfigurationModeFrequencyMins" : "<value>"
}
```

PowerShell format

```

$privateConfig = '{
    "RegistrationUrl": "<azure-automation-account-url>",
    "RegistrationKey": "<azure-automation-account-key>"
}'

$publicConfig = '{
    "ExtensionAction" : "Register",
    "NodeConfigurationName": "<node-configuration-name>",
    "RefreshFrequencyMins": "<value>",
    "ConfigurationMode": "<ApplyAndMonitor | ApplyAndAutoCorrect | ApplyOnly>",
    "ConfigurationModeFrequencyMins": "<value>"
}'

```

### Apply an MOF configuration file (in an Azure storage account) to the VM

protected.json

```
{
    "StorageAccountName": "<storage-account-name>",
    "StorageAccountKey": "<storage-account-key>"
}
```

public.json

```
{
    "FileUri": "<mof-file-uri>",
    "ExtensionAction": "Push"
}
```

PowerShell format

```

$privateConfig = '{
    "StorageAccountName": "<storage-account-name>",
    "StorageAccountKey": "<storage-account-key>"
}'

$publicConfig = '{
    "FileUri": "<mof-file-uri>",
    "ExtensionAction": "Push"
}'

```

### Apply an MOF configuration file (in public storage) to the VM

public.json

```
{
    "FileUri": "<mof-file-uri>"
}
```

PowerShell format

```

$publicConfig = '{
    "FileUri": "<mof-file-uri>"
}'

```

### Apply a meta MOF configuration file (in an Azure storage account) to the VM

protected.json

```
{  
  "StorageAccountName": "<storage-account-name>",  
  "StorageAccountKey": "<storage-account-key>"  
}
```

public.json

```
{  
  "ExtensionAction": "Pull",  
  "FileUri": "<meta-mof-file-uri>"  
}
```

PowerShell format

```
$privateConfig = '{  
  "StorageAccountName": "<storage-account-name>",  
  "StorageAccountKey": "<storage-account-key>"  
}'  
  
$publicConfig = '{  
  "ExtensionAction": "Pull",  
  "FileUri": "<meta-mof-file-uri>"  
}'
```

## Apply a meta MOF configuration file (in public storage) to the VM

public.json

```
{  
  "FileUri": "<meta-mof-file-uri>",  
  "ExtensionAction": "Pull"  
}
```

PowerShell format

```
$publicConfig = '{  
  "FileUri": "<meta-mof-file-uri>",  
  "ExtensionAction": "Pull"  
}'
```

## Install a custom resource module (a zip file in an Azure storage account) to the VM

protected.json

```
{  
  "StorageAccountName": "<storage-account-name>",  
  "StorageAccountKey": "<storage-account-key>"  
}
```

public.json

```
{  
  "ExtensionAction": "Install",  
  "FileUri": "<resource-zip-file-uri>"  
}
```

PowerShell format

```
$privateConfig = '{
  "StorageAccountName": "<storage-account-name>",
  "StorageAccountKey": "<storage-account-key>"
}'

$publicConfig = '{
  "ExtensionAction": "Install",
  "FileUri": "<resource-zip-file-uri>"
}'
```

### Install a custom resource module (a zip file in public storage) to the VM

public.json

```
{
  "ExtensionAction": "Install",
  "FileUri": "<resource-zip-file-uri>"
}
```

PowerShell format

```
$publicConfig = '{
  "ExtensionAction": "Install",
  "FileUri": "<resource-zip-file-uri>"
}'
```

### Remove a custom resource module from the VM

public.json

```
{
  "ResourceName": "<resource-name>",
  "ExtensionAction": "Remove"
}
```

PowerShell format

```
$publicConfig = '{
  "ResourceName": "<resource-name>",
  "ExtensionAction": "Remove"
}'
```

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when you deploy one or more virtual machines that require post-deployment configuration, such as onboarding to Azure Automation.

The sample Resource Manager template is [dsc-linux-azure-storage-on-ubuntu](#) and [dsc-linux-public-storage-on-ubuntu](#).

For more information about the Azure Resource Manager template, see [Authoring Azure Resource Manager templates](#).

# Azure CLI deployment

## Use [Azure CLI][azure-cli]

Before you deploy the DSCForLinux extension, configure your `public.json` and `protected.json` according to the different scenarios in section 3.

### Classic

#### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

The classic deployment mode is also called Azure Service Management mode. You can switch to it by running:

```
$ azure config mode asm
```

You can deploy the DSCForLinux extension by running:

```
$ azure vm extension set <vm-name> DSCForLinux Microsoft.OSTCExtensions <version> \
--private-config-path protected.json --public-config-path public.json
```

To learn the latest extension version available, run:

```
$ azure vm extension list
```

### Resource Manager

You can switch to Azure Resource Manager mode by running:

```
$ azure config mode arm
```

You can deploy the DSCForLinux extension by running:

```
$ azure vm extension set <resource-group> <vm-name> \
DSCForLinux Microsoft.OSTCExtensions <version> \
--private-config-path protected.json --public-config-path public.json
```

#### NOTE

In Azure Resource Manager mode, `azure vm extension list` isn't available for now.

## Use [Azure PowerShell][azure-powershell]

### Classic

You can sign in to your Azure account in Azure Service Management mode by running:

```
Add-AzureAccount
```

And deploy the DSCForLinux extension by running:

```
$vmname = '<vm-name>'  
$vm = Get-AzureVM -ServiceName $vmname -Name $vmname  
$extensionName = 'DSCForLinux'  
$publisher = 'Microsoft.OSTCExtensions'  
$version = '< version>'
```

Change the content of \$privateConfig and \$publicConfig according to different scenarios in the previous section.

```
$privateConfig = '{  
    "StorageAccountName": "<storage-account-name>",  
    "StorageAccountKey": "<storage-account-key>"  
'
```

```
$publicConfig = '{  
    "ExtensionAction": "Push",  
    "FileUri": "<mof-file-uri>"  
'
```

```
Set-AzureVMExtension -ExtensionName $extensionName -VM $vm -Publisher $publisher `  
-Version $version -PrivateConfiguration $privateConfig `  
-PublicConfiguration $publicConfig | Update-AzureVM
```

## Resource Manager

You can sign in to your Azure account in Azure Resource Manager mode by running:

```
Login-AzAccount
```

To learn more about how to use Azure PowerShell with Azure Resource Manager, see [Manage Azure resources by using Azure PowerShell](#).

You can deploy the DSCForLinux extension by running:

```
$rgName = '<resource-group-name>'  
$vmName = '<vm-name>'  
$location = '< location>'  
$extensionName = 'DSCForLinux'  
$publisher = 'Microsoft.OSTCExtensions'  
$version = '< version>'
```

Change the content of \$privateConfig and \$publicConfig according to different scenarios in the previous section.

```
$privateConfig = '{  
    "StorageAccountName": "<storage-account-name>",  
    "StorageAccountKey": "<storage-account-key>"  
'
```

```
$publicConfig = '{
    "ExtensionAction": "Push",
    "FileUri": "<mof-file-uri>"
}'
```

```
Set-AzVMExtension -ResourceGroupName $rgName -VMName $vmName -Location $location ` 
    -Name $extensionName -Publisher $publisher -ExtensionType $extensionName ` 
    -TypeHandlerVersion $version -SettingString $publicConfig -ProtectedSettingString $privateConfig
```

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command by using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension execution output is logged to the following file:

```
/var/log/azure/<extension-name>/<version>/extension.log file.
```

Error code: 51 represents either unsupported distribution or unsupported extension action. In some cases, DSC Linux extension fails to install OMI when a higher version of OMI already exists in the machine. [error response: (000003)Downgrade not allowed]

### Support

If you need more help at any point in this article, contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure Support incident. Go to the [Azure Support site](#), and select **Get support**. For information about using Azure Support, read the [Microsoft Azure Support FAQ](#).

## Next steps

For more information about extensions, see [Virtual machine extensions and features for Linux](#).

# PowerShell DSC Extension

9/21/2022 • 5 minutes to read • [Edit Online](#)

## Overview

The PowerShell DSC Extension for Windows is published and supported by Microsoft. The extension uploads and applies a PowerShell DSC Configuration on an Azure VM. The DSC Extension calls into PowerShell DSC to enact the received DSC configuration on the VM. This document details the supported platforms, configurations, and deployment options for the DSC virtual machine extension for Windows.

### NOTE

Before you enable the DSC extension, we would like you to know that a newer version of DSC is now available in preview, managed by a feature of Azure Policy named [guest configuration](#). The guest configuration feature combines features of the Desired State Configuration (DSC) extension handler, Azure Automation State Configuration, and the most commonly requested features from customer feedback. Guest configuration also includes hybrid machine support through [Arc-enabled servers](#).

## Prerequisites

### Operating system

The DSC Extension supports the following OS's

Windows Server 2019, Windows Server 2016, Windows Server 2012R2, Windows Server 2012, Windows Server 2008 R2 SP1, Windows Client 7/8.1/10

### Internet connectivity

The DSC extension for Windows requires that the target virtual machine is able to communicate with Azure and the location of the configuration package (.zip file) if it is stored in a location outside of Azure.

## Extension schema

The following JSON shows the schema for the settings portion of the DSC Extension in an Azure Resource Manager template.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "Microsoft.Powershell.DSC",
  "apiVersion": "2018-10-01",
  "location": "<location>",
  "properties": {
    "publisher": "Microsoft.Powershell",
    "type": "DSC",
    "typeHandlerVersion": "2.77",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "wmfVersion": "latest",
      "configuration": {
        "url": "http://validURLToConfigLocation",
        "script": "ConfigurationScript.ps1",
        "function": "ConfigurationFunction"
      },
      "configurationArguments": {
        "argument1": "Value1",
        "argument2": "Value2"
      },
      "configurationData": {
        "url": "https://foo.psd1"
      },
      "privacy": {
        "dataCollection": "enable"
      },
      "advancedOptions": {
        "forcePullAndApply": false,
        "downloadMappings": {
          "specificDependencyKey": "https://myCustomDependencyLocation"
        }
      }
    },
    "protectedSettings": {
      "configurationArguments": {
        "parameterOfTypePSCredential1": {
          "userName": "UsernameValue1",
          "password": "PasswordValue1"
        },
        "parameterOfTypePSCredential2": {
          "userName": "UsernameValue2",
          "password": "PasswordValue2"
        }
      },
      "configurationUrlSasToken": "?g!bber1sht0k3n",
      "configurationDataUrlSasToken": "?dataAcC355T0k3N"
    }
  }
}
```

## Property values

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2018-10-01	date
publisher	Microsoft.Powershell.DSC	string
type	DSC	string
typeHandlerVersion	2.77	int

## Settings Property values

NAME	DATA TYPE	DESCRIPTION
settings.wmfVersion	string	Specifies the version of the Windows Management Framework that should be installed on your VM. Setting this property to 'latest' will install the most updated version of WMF. The only current possible values for this property are '4.0', '5.0', and 'latest'. These possible values are subject to updates. The default value is 'latest'.
settings.configuration.url	string	Specifies the URL location from which to download your DSC configuration zip file. If the URL provided requires a SAS token for access, you will need to set the <code>protectedSettings.configurationUrlSasToken</code> property to the value of your SAS token. This property is required if <code>settings.configuration.script</code> and/or <code>settings.configuration.function</code> are defined.
settings.configuration.script	string	Specifies the file name of the script that contains the definition of your DSC configuration. This script must be in the root folder of the zip file downloaded from the URL specified by the <code>configuration.url</code> property. This property is required if <code>settings.configuration.url</code> and/or <code>settings.configuration.script</code> are defined.
settings.configuration.function	string	Specifies the name of your DSC configuration. The configuration named must be contained in the script defined by <code>configuration.script</code> . This property is required if <code>settings.configuration.url</code> and/or <code>settings.configuration.function</code> are defined.
settings.configurationArguments	Collection	Defines any parameters you would like to pass to your DSC configuration. This property will not be encrypted.
settings.configurationData.url	string	Specifies the URL from which to download your configuration data (.pds1) file to use as input for your DSC configuration. If the URL provided requires a SAS token for access, you will need to set the <code>protectedSettings.configurationDataUrlSasToken</code> property to the value of your SAS token.

NAME	DATA TYPE	DESCRIPTION
settings.privacy.dataEnabled	string	Enables or disables telemetry collection. The only possible values for this property are 'Enable', 'Disable', "", or \$null. Leaving this property blank or null will enable telemetry
settings.advancedOptions.forcePullAndApply	Bool	This setting is designed to enhance the experience of working with the extension to register nodes with Azure Automation DSC. If the value is <code>\$true</code> , the extension will wait for the first run of the configuration pulled from the service before returning success/failure. If the value is set to <code>\$false</code> , the status returned by the extension will only refer to whether the node was registered with Azure Automation State Configuration successfully and the node configuration will not be run during the registration.
settings.advancedOptions.downloadMappings	Collection	Defines alternate locations to download dependencies such as WMF and .NET

### Protected Settings Property values

NAME	DATA TYPE	DESCRIPTION
protectedSettings.configurationArguments	string	Defines any parameters you would like to pass to your DSC configuration. This property will be encrypted.
protectedSettings.configurationUrlSasToken	string	Specifies the SAS token to access the URL defined by configuration.url. This property will be encrypted.
protectedSettings.configurationDataUrlSasToken	string	Specifies the SAS token to access the URL defined by configurationData.url. This property will be encrypted.

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates are ideal when deploying one or more virtual machines that require post deployment configuration. A sample Resource Manager template that includes the DSC extension for Windows can be found on the [Azure Quick Start Gallery](#).

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

Extension package is downloaded and deployed to this location on the Azure VM

```
C:\Packages\Plugins\{Extension_Name}\{Extension_Version}
```

Extension status file contains the sub status and status success/error codes along with the detailed error and description for each extension run.

```
C:\Packages\Plugins\{Extension_Name}\{Extension_Version}\Status\{0}.Status -> {0} being the sequence number
```

Extension output logs are logged to the following directory:

```
C:\WindowsAzure\Logs\Plugins\{Extension_Name}\{Extension_Version}
```

### Error codes and their meanings

ERROR CODE	MEANING	POSSIBLE ACTION
1000	Generic error	The message for this error is provided by the specific exception in extension logs
52	Extension Install Error	The message for this error is provided by the specific exception
1002	Wmf Install Error	Error while installing WMF.
1004	Invalid Zip Package	Invalid zip ; Error unpacking the zip
1100	Argument Error	Indicates a problem in the input provided by the user. The message for the error is provided by the specific exception

### Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Pass credentials to the Azure DSCExtension handler

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article covers the Desired State Configuration (DSC) extension for Azure. For an overview of the DSC extension handler, see [Introduction to the Azure Desired State Configuration extension handler](#).

## Pass in credentials

As part of the configuration process, you might need to set up user accounts, access services, or install a program in a user context. To do these things, you need to provide credentials.

You can use DSC to set up parameterized configurations. In a parameterized configuration, credentials are passed into the configuration and securely stored in .mof files. The Azure extension handler simplifies credential management by providing automatic management of certificates.

The following DSC configuration script creates a local user account with the specified password:

```
configuration Main
{
    param(
        [Parameter(Mandatory=$true)]
        [ValidateNotNullOrEmpty()]
        [PsCredential]
        $Credential
    )
    Node localhost {
        User LocalUserAccount
        {
            Username = $Credential.UserName
            Password = $Credential
            Disabled = $false
            Ensure = "Present"
            FullName = "Local User Account"
            Description = "Local User Account"
            PasswordNeverExpires = $true
        }
    }
}
```

It's important to include **node localhost** as part of the configuration. The extension handler specifically looks for the **node localhost** statement. If this statement is missing, the following steps don't work. It's also important to include the typecast **[PsCredential]**. This specific type triggers the extension to encrypt the credential.

To publish this script to Azure Blob storage:

```
Publish-AzVMDscConfiguration -ConfigurationPath .\user_configuration.ps1
```

To set the Azure DSC extension and provide the credential:

```
$configurationName = 'Main'  
$configurationArguments = @{ Credential = Get-Credential }  
$configurationArchive = 'user_configuration.ps1.zip'  
$vm = Get-AzVM -Name 'example-1'  
  
$vm = Set-AzVMDscExtension -VMName $vm -ConfigurationArchive $configurationArchive -ConfigurationName  
$configurationName -ConfigurationArgument @configurationArguments  
  
$vm | Update-AzVM
```

## How a credential is secured

Running this code prompts for a credential. After the credential is provided, it's briefly stored in memory. When the credential is published by using the **Set-AzVMDscExtension** cmdlet, the credential is transmitted over HTTPS to the VM. In the VM, Azure stores the credential encrypted on disk by using the local VM certificate. The credential is briefly decrypted in memory, and then it's re-encrypted to pass it to DSC.

This process is different than [using secure configurations without the extension handler](#). The Azure environment gives you a way to transmit configuration data securely via certificates. When you use the DSC extension handler, you don't need to provide **\$CertificatePath** or a **\$CertificateID**/ **\$Thumbprint** entry in **ConfigurationData**.

## Next steps

- Get an [introduction to Azure DSC extension handler](#).
- Examine the [Azure Resource Manager template for the DSC extension](#).
- For more information about PowerShell DSC, go to the [PowerShell documentation center](#).
- For more functionality that you can manage by using PowerShell DSC, and for more DSC resources, browse the [PowerShell gallery](#).

# Desired State Configuration extension with Azure Resource Manager templates

9/21/2022 • 10 minutes to read • [Edit Online](#)

This article describes the Azure Resource Manager template for the Desired State Configuration (DSC) extension handler. Many of the examples use **RegistrationURL** (provided as a String) and **RegistrationKey** (provided as a **PSCredential**) to onboard with Azure Automation. For details about obtaining those values, see [Use DSC metaconfiguration to register hybrid machines](#).

## NOTE

You might encounter slightly different schema examples. The change in schema occurred in the October 2016 release. For details, see [Update from a previous format](#).

Before you enable the DSC extension, we would like you to know that a newer version of DSC is now available in preview, managed by a feature of Azure Policy named [guest configuration](#). The guest configuration feature combines features of the Desired State Configuration (DSC) extension handler, Azure Automation State Configuration, and the most commonly requested features from customer feedback. Guest configuration also includes hybrid machine support through [Arc-enabled servers](#).

## Template example for a Windows VM

The following snippet goes in the **Resource** section of the template. The DSC extension inherits default extension properties. For more information, see [VirtualMachineExtension class](#).

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "[concat(parameters('VMName'), '/Microsoft.Powershell.DSC')]",
  "apiVersion": "2018-06-01",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', parameters('VMName'))]"
  ],
  "properties": {
    "publisher": "Microsoft.Powershell",
    "type": "DSC",
    "typeHandlerVersion": "2.77",
    "autoUpgradeMinorVersion": true,
    "protectedSettings": {
      "Items": {
        "registrationKeyPrivate": "[listKeys(resourceId('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName')), '2018-06-30').Keys[0].value]"
      }
    },
    "settings": {
      "Properties": [
        {
          "Name": "RegistrationKey",
          "Value": {
            "UserName": "PLACEHOLDER_DONOTUSE",
            "Password": "PrivateSettingsRef:registrationKeyPrivate"
          },
          "TypeName": "System.Management.Automation.PSCredential"
        },
        {
          "Name": "RegistrationUrl",
          "Value": "[reference(concat('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName'))).registrationUrl]",
          "TypeName": "System.String"
        },
        {
          "Name": "NodeConfigurationName",
          "Value": "[parameters('nodeConfigurationName')]",
          "TypeName": "System.String"
        }
      ]
    }
  }
}
```

## Template example for Windows virtual machine scale sets

A virtual machine scale set node has a **properties** section that has a **VirtualMachineProfile**, **extensionProfile** attribute. Under **extensions**, add the details for DSC Extension.

The DSC extension inherits default extension properties. For more information, see [VirtualMachineScaleSetExtension class](#).

```

"extensionProfile": {
  "extensions": [
    {
      "name": "Microsoft.Powershell.DSC",
      "properties": {
        "publisher": "Microsoft.Powershell",
        "type": "DSC",
        "typeHandlerVersion": "2.77",
        "autoUpgradeMinorVersion": true,
        "protectedSettings": {
          "Items": [
            "registrationKeyPrivate": "[listKeys(resourceId('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName')), '2018-06-30').Keys[0].value]"
          ]
        },
        "settings": {
          "Properties": [
            {
              "Name": "RegistrationKey",
              "Value": {
                "UserName": "PLACEHOLDER_DONOTUSE",
                "Password": "PrivateSettingsRef:registrationKeyPrivate"
              },
              "TypeName": "System.Management.Automation.PSCredential"
            },
            {
              "Name": "RegistrationUrl",
              "Value": "[reference(concat('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName'))).registrationUrl]",
              "TypeName": "System.String"
            },
            {
              "Name": "NodeConfigurationName",
              "Value": "[parameters('nodeConfigurationName')]",
              "TypeName": "System.String"
            }
          ]
        }
      }
    ]
  }
}

```

## Detailed settings information

Use the following schema in the **settings** section of the Azure DSC extension in a Resource Manager template.

For a list of the arguments that are available for the default configuration script, see [Default configuration script](#).

```

"settings": {
    "wmfVersion": "latest",
    "configuration": {
        "url": "http://validURLToConfigLocation",
        "script": "ConfigurationScript.ps1",
        "function": "ConfigurationFunction"
    },
    "configurationArguments": {
        "argument1": "Value1",
        "argument2": "Value2"
    },
    "configurationData": {
        "url": "https://foo.psd1"
    },
    "privacy": {
        "dataCollection": "enable"
    },
    "advancedOptions": {
        "downloadMappings": {
            "customWmfLocation": "http://myWMFlocation"
        }
    }
},
"protectedSettings": {
    "configurationArguments": {
        "parameterOfTypePSCredential1": {
            "userName": "UsernameValue1",
            "password": "PasswordValue1"
        },
        "parameterOfTypePSCredential2": {
            "userName": "UsernameValue2",
            "password": "PasswordValue2"
        }
    },
    "configurationUrlSasToken": "?g!bber1sh0k3n",
    "configurationDataUrlSasToken": "?dataAcc355T0k3N"
}
}

```

## Details

PROPERTY NAME	TYPE	DESCRIPTION
settings.wmfVersion	string	Specifies the version of Windows Management Framework (WMF) that should be installed on your VM. Setting this property to <b>latest</b> installs the most recent version of WMF. Currently, the only possible values for this property are <b>4.0</b> , <b>5.0</b> , <b>5.1</b> , and <b>latest</b> . These possible values are subject to updates. The default value is <b>latest</b> .

PROPERTY NAME	TYPE	DESCRIPTION
settings.configuration.url	string	Specifies the URL location from which to download your DSC configuration .zip file. If the URL provided requires an SAS token for access, set the <b>protectedSettings.configurationUrlIsSasToken</b> property to the value of your SAS token. This property is required if <b>settings.configuration.script</b> or <b>settings.configuration.function</b> are defined. If no value is given for these properties, the extension calls the default configuration script to set Location Configuration Manager (LCM) metadata, and arguments should be supplied.
settings.configuration.script	string	Specifies the file name of the script that contains the definition of your DSC configuration. This script must be in the root folder of the .zip file that's downloaded from the URL specified by the <b>settings.configuration.url</b> property. This property is required if <b>settings.configuration.url</b> or <b>settings.configuration.script</b> are defined. If no value is given for these properties, the extension calls the default configuration script to set LCM metadata, and arguments should be supplied.
settings.configuration.function	string	Specifies the name of your DSC configuration. The configuration that is named must be included in the script that <b>settings.configuration.script</b> defines. This property is required if <b>settings.configuration.url</b> or <b>settings.configuration.function</b> are defined. If no value is given for these properties, the extension calls the default configuration script to set LCM metadata, and arguments should be supplied.
settings.configurationArguments	Collection	Defines any parameters that you want to pass to your DSC configuration. This property is not encrypted.
settings.configurationData.url	string	Specifies the URL from which to download your configuration data (.psd1) file to use as input for your DSC configuration. If the URL provided requires an SAS token for access, set the <b>protectedSettings.configurationDataUrlIsSasToken</b> property to the value of your SAS token.

PROPERTY NAME	TYPE	DESCRIPTION
settings.privacy.dataCollection	string	Enables or disables telemetry collection. The only possible values for this property are <b>Enable</b> , <b>Disable</b> , "", or <b>\$null</b> . Leaving this property blank or null enables telemetry. The default value is "". For more information, see <a href="#">Azure DSC extension data collection</a> .
settings.advancedOptions.downloadMappings	Collection	Defines alternate locations from which to download WMF. For more information, see <a href="#">Azure DSC extension 2.8 and how to map downloads of the extension dependencies to your own location</a> .
protectedSettings.configurationArguments	Collection	Defines any parameters that you want to pass to your DSC configuration. This property is encrypted.
protectedSettings.configurationUrlSasToken	string	Specifies the SAS token to use to access the URL that <b>settings.configuration.url</b> defines. This property is encrypted.
protectedSettings.configurationDataUrlSasToken	string	Specifies the SAS token to use to access the URL that <b>settings.configurationData.url</b> defines. This property is encrypted.

## Default configuration script

For more information about the following values, see [Local Configuration Manager basic settings](#). You can use the DSC extension default configuration script to configure only the LCM properties that are listed in the following table.

PROPERTY NAME	TYPE	DESCRIPTION
protectedSettings.configurationArguments.RegistrationKey	PSCredential	Required property. Specifies the key that's used for a node to register with the Azure Automation service as the password of a PowerShell credential object. This value can be automatically discovered by using the <b>listkeys</b> method against the Automation account. See the <a href="#">example</a> .
settings.configurationArguments.RegistrationUrl	string	Required property. Specifies the URL of the Automation endpoint where the node attempts to register. This value can be automatically discovered by using the <b>reference</b> method against the Automation account.
settings.configurationArguments.NodeConfigurationName	string	Required property. Specifies the node configuration in the Automation account to assign to the node.

PROPERTY NAME	TYPE	DESCRIPTION
settings.configurationArguments.ConfigurationMode	string	Specifies the mode for LCM. Valid options include <b>ApplyOnly</b> , <b>ApplyandMonitor</b> , and <b>ApplyandAutoCorrect</b> . The default value is <b>ApplyandMonitor</b> .
settings.configurationArguments.RefreshFrequencyMins	uint32	Specifies how often LCM attempts to check with the Automation account for updates. Default value is <b>30</b> . Minimum value is <b>15</b> .
settings.configurationArguments.ConfigurationModeFrequencyMins	uint32	Specifies how often LCM validates the current configuration. Default value is <b>15</b> . Minimum value is <b>15</b> .
settings.configurationArguments.RebootNodeIfNeeded	boolean	Specifies whether a node can be automatically rebooted if a DSC operation requests it. Default value is <b>false</b> .
settings.configurationArguments.ActionAfterReboot	string	Specifies what happens after a reboot when applying a configuration. Valid options are <b>ContinueConfiguration</b> and <b>StopConfiguration</b> . Default value is <b>ContinueConfiguration</b> .
settings.configurationArguments.AllowModuleOverwrite	boolean	Specifies whether LCM overwrites existing modules on the node. Default value is <b>false</b> .

## settings vs. protectedSettings

All settings are saved in a settings text file on the VM. Properties listed under **settings** are public properties. Public properties aren't encrypted in the settings text file. Properties listed under **protectedSettings** are encrypted with a certificate and are not shown in plain text in the settings file on the VM.

If the configuration needs credentials, you can include the credentials in **protectedSettings**:

```
"protectedSettings": {
    "configurationArguments": {
        "parameterOfTypePSCredential1": {
            "userName": "UsernameValue1",
            "password": "PasswordValue1"
        }
    }
}
```

## Example configuration script

The following example shows the default behavior for the DSC extension, which is to provide metadata settings to LCM and register with the Automation DSC service. Configuration arguments are required. Configuration arguments are passed to the default configuration script to set LCM metadata.

```

"settings": {
    "configurationArguments": {
        "RegistrationUrl" : "[parameters('registrationUrl1')]",
        "NodeConfigurationName" : "nodeConfigurationNameValue1"
    }
},
"protectedSettings": {
    "configurationArguments": {
        "RegistrationKey": {
            "userName": "NOT_USED",
            "Password": "registrationKey"
        }
    }
}
}

```

## Example using the configuration script in Azure Storage

The following example is from the [DSC extension handler overview](#). This example uses Resource Manager templates instead of cmdlets to deploy the extension. Save the lisInstall.ps1 configuration, place it in a .zip file (example: `iisinstall.zip`), and then upload the file in an accessible URL. This example uses Azure Blob storage, but you can download .zip files from any arbitrary location.

In the Resource Manager template, the following code instructs the VM to download the correct file, and then run the appropriate PowerShell function:

```

"settings": {
    "configuration": {
        "url": "https://demo.blob.core.windows.net/iisinstall.zip",
        "script": "IisInstall.ps1",
        "function": "IISInstall"
    }
},
"protectedSettings": {
    "configurationUrlsSasToken": "odLPL/U1p9lvcnp..."
}

```

## Example using referenced Azure Automation registration values

The following example gets the **RegistrationUrl** and **RegistrationKey** by referencing the Azure Automation account properties and using the `listkeys` method to retrieve the Primary Key (0). In this example, the parameters **automationAccountName** and **NodeConfigName** were provided to the template.

```

"settings": {
    "RegistrationUrl" : "[reference(concat('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName'))).registrationUrl]",
    "NodeConfigurationName" : "[parameters('NodeConfigName')]"
},
"protectedSettings": {
    "configurationArguments": {
        "RegistrationKey": {
            "userName": "NOT_USED",
            "Password": "[listKeys(resourceId('Microsoft.Automation/automationAccounts/',
parameters('automationAccountName')), '2018-01-15').Keys[0].value]"
        }
    }
}

```

## Update from a previous format

Any settings in a previous format of the extension (and which have the public properties **ModulesUrl**, **ModuleSource**, **ModuleVersion**, **ConfigurationFunction**, **SasToken**, or **Properties**) automatically adapt to the current format of the extension. They run just as they did before.

The following schema shows what the previous settings schema looked like:

```
"settings": {
    "WMFVersion": "latest",
    "ModulesUrl": "https://UrlToZipContainingConfigurationScript.ps1.zip",
    "SasToken": "SAS Token if ModulesUrl points to private Azure Blob Storage",
    "ConfigurationFunction": "ConfigurationScript.ps1\\ConfigurationFunction",
    "Properties": {
        "ParameterToConfigurationFunction1": "Value1",
        "ParameterToConfigurationFunction2": "Value2",
        "ParameterOfTypePSCredential1": {
            "UserName": "UsernameValue1",
            "Password": "PrivateSettingsRef:Key1"
        },
        "ParameterOfTypePSCredential2": {
            "UserName": "UsernameValue2",
            "Password": "PrivateSettingsRef:Key2"
        }
    }
},
"protectedSettings": {
    "Items": {
        "Key1": "PasswordValue1",
        "Key2": "PasswordValue2"
    },
    "DataBlobUri": "https://UrlToConfigurationDataWithOptionalSasToken.psd1"
}
```

Here's how the previous format adapts to the current format:

CURRENT PROPERTY NAME	PREVIOUS SCHEMA EQUIVALENT
settings.wmfVersion	settings.WMFVersion
settings.configuration.url	settings.ModulesUrl
settings.configuration.script	First part of settings.ConfigurationFunction (before \\)
settings.configuration.function	Second part of settings.ConfigurationFunction (after \\)
settings.configuration.module.name	settings.ModuleSource
settings.configuration.module.version	settings.ModuleVersion
settings.configurationArguments	settings.Properties
settings.configurationData.url	protectedSettings.DataBlobUri (without SAS token)
settings.privacy.dataCollection	settings.Privacy.dataCollection
settings.advancedOptions.downloadMappings	settings.AdvancedOptions.DownloadMappings

CURRENT PROPERTY NAME	PREVIOUS SCHEMA EQUIVALENT
protectedSettings.configurationArguments	protectedSettings.Properties
protectedSettings.configurationUrlSasToken	settings.SasToken
protectedSettings.configurationDataUrlSasToken	SAS token from protectedSettings.DataBlobUri

## Troubleshooting

Here are some of the errors you might run into and how you can fix them.

### Invalid values

"Privacy.dataCollection is '{0}'. The only possible values are '', 'Enable', and 'Disable'". "WmfVersion is '{0}'. Only possible values are ... and 'latest'".

**Problem:** A provided value is not allowed.

**Solution:** Change the invalid value to a valid value. For more information, see the table in [Details](#).

### Invalid URL

"ConfigurationData.url is '{0}'. This is not a valid URL" "DataBlobUri is '{0}'. This is not a valid URL"

"Configuration.url is '{0}'. This is not a valid URL"

**Problem:** A provided URL is not valid.

**Solution:** Check all your provided URLs. Ensure that all URLs resolve to valid locations that the extension can access on the remote machine.

### Invalid RegistrationKey type

"Invalid type for parameter RegistrationKey of type PSCredential."

**Problem:** The *RegistrationKey* value in protectedSettings.configurationArguments cannot be provided as any type other than a PSCredential.

**Solution:** Change your protectedSettings.configurationArguments entry for RegistrationKey to a PSCredential type using the following format:

```

"configurationArguments": {
    "RegistrationKey": {
        "userName": "NOT_USED",
        "Password": "RegistrationKey"
    }
}

```

### Invalid ConfigurationArgument type

"Invalid configurationArguments type {0}"

**Problem:** The *ConfigurationArguments* property can't resolve to a **Hash table** object.

**Solution:** Make your *ConfigurationArguments* property a **Hash table**. Follow the format provided in the preceding examples. Watch for quotes, commas, and braces.

### Duplicate ConfigurationArguments

"Found duplicate arguments '{0}' in both public and protected configurationArguments"

**Problem:** The *ConfigurationArguments* in public settings and the *ConfigurationArguments* in protected settings

have properties with the same name.

**Solution:** Remove one of the duplicate properties.

### **Missing properties**

"settings.Configuration.function requires that settings.configuration.url or settings.configuration.module is specified"

"settings.Configuration.url requires that settings.configuration.script is specified"

"settings.Configuration.script requires that settings.configuration.url is specified"

"settings.Configuration.url requires that settings.configuration.function is specified"

"protectedSettings.ConfigurationUrlSasToken requires that settings.configuration.url is specified"

"protectedSettings.ConfigurationDataUrlSasToken requires that settings.configurationData.url is specified"

**Problem:** A defined property needs another property, which is missing.

**Solutions:**

- Provide the missing property.
- Remove the property that needs the missing property.

## Next steps

- Learn about [using virtual machine scale sets with the Azure DSC extension](#).
- Find more details about [DSC's secure credential management](#).
- Get an [introduction to the Azure DSC extension handler](#).
- For more information about PowerShell DSC, go to the [PowerShell documentation center](#).

# Manage administrative users, SSH, and check or repair disks on Linux VMs using the VMAccess Extension with the Azure CLI

9/21/2022 • 5 minutes to read • [Edit Online](#)

## Overview

The disk on your Linux VM is showing errors. You somehow reset the root password for your Linux VM or accidentally deleted your SSH private key. If that happened back in the days of the datacenter, you would need to drive there and then open the KVM to get at the server console. Think of the Azure VMAccess extension as that KVM switch that allows you to access the console to reset access to Linux or perform disk level maintenance.

This article shows you how to use the Azure VMAccess Extension to check or repair a disk, reset user access, manage administrative user accounts, or update the SSH configuration on Linux when they are running as Azure Resource Manager virtual machines. If you need to manage Classic virtual machines - you can follow the instructions found in the [classic VM documentation](#).

### NOTE

If you use the VMAccess Extension to reset the password of your VM after installing the AAD Login Extension you will need to rerun the AAD Login Extension to re-enable AAD Login for your machine.

## Prerequisites

### Operating system

The VM Access extension can be run against these Linux distributions:

DISTRIBUTION	VERSION
Ubuntu	16.04 LTS, 14.04 LTS and 12.04 LTS
Debian	Debian 7.9+, 8.2+
Red Hat	RHEL 6.7+, 7.1+
Oracle Linux	6.4+, 7.0+
Suse	11 and 12
OpenSuse	openSUSE Leap 42.2+
CentOS	CentOS 6.3+, 7.0+
CoreOS	494.4.0+

## Ways to use the VMAccess Extension

There are two ways that you can use the VMAccess Extension on your Linux VMs:

- Use the Azure CLI and the required parameters.
- [Use raw JSON files that the VMAccess Extension process](#) and then act on.

The following examples use `az vm user` commands. To perform these steps, you need the latest [Azure CLI](#) installed and logged in to an Azure account using `az login`.

## Update SSH key

The following example updates the SSH key for the user `azureuser` on the VM named `myVM`:

```
az vm user update \
--resource-group myResourceGroup \
--name myVM \
--username azureuser \
--ssh-key-value ~/.ssh/id_rsa.pub
```

**NOTE:** The `az vm user update` command appends the new public key text to the `~/.ssh/authorized_keys` file for the admin user on the VM. This does not replace or remove any existing SSH keys. This will not remove prior keys set at deployment time or subsequent updates via the VMAccess Extension.

## Reset password

The following example resets the password for the user `azureuser` on the VM named `myVM`:

```
az vm user update \
--resource-group myResourceGroup \
--name myVM \
--username azureuser \
--password myNewPassword
```

## Restart SSH

The following example restarts the SSH daemon and resets the SSH configuration to default values on a VM named `myVM`:

```
az vm user reset-ssh \
--resource-group myResourceGroup \
--name myVM
```

## Create an administrative/sudo user

The following example creates a user named `myNewUser` with **sudo** permissions. The account uses an SSH key for authentication on the VM named `myVM`. This method is designed to help you regain access to a VM in the event that current credentials are lost or forgotten. As a best practice, accounts with **sudo** permissions should be limited.

```
az vm user update \
--resource-group myResourceGroup \
--name myVM \
--username myNewUser \
--ssh-key-value ~/.ssh/id_rsa.pub
```

# Delete a user

The following example deletes a user named `myNewUser` on the VM named `myVM`:

```
az vm user delete \
--resource-group myResourceGroup \
--name myVM \
--username myNewUser
```

# Use JSON files and the VMAccess Extension

The following examples use raw JSON files. Use [az vm extension set](#) to then call your JSON files. These JSON files can also be called from Azure templates.

## Reset user access

If you have lost access to root on your Linux VM, you can launch a VMAccess script to update a user's SSH key or password.

To update the SSH public key of a user, create a file named `update_ssh_key.json` and add settings in the following format. Replace `username` and `ssh_key` with your own information:

```
{
  "username": "azureuser",
  "ssh_key": "ssh-rsa"
  AAAAB3NzaC1yc2EAAAQABAAQACZ3S7gGp3rcbKmG2Y4vGZFMuMZCwoUzZNGxxxxxx2XV2x9FfAh8igD+1F8UdjFX3t5ebMm6BnnMh8
  fHwkTRd0t3LDQq8o8ElTBrzAKPxZN2tMzn0Ds5H1emb2UX0oRIGRcvWqsd4oJmxsXa/Si98Wa6RHwbc9Qzhw80KAc0VhmndZAZAGR+Wq6ys
  1No5TMOr1/ZyQAook5C4FtcSGn3Y+WczaogWIxG4ZaWk128g79VIeJcIQqOjPodHvQAh117qD1ItVvBFM0ben3GyhYTm7k4Yw1Edk0Nm4yV/
  UIW0la1rmyztSBQIm9sZmSq44XXgjVmDHNF8UFcz1ToE4r2SdwTmZv00T2i5faeYnHzxiLPA3Enub7xxxxxxxxFArnqad7M01SY1kLemhX9eF
  jLNW4mJe56Fu4NiWjkR9APSZqrYeKaqr4KUC68QpVasNjhbxPSf/PcjF3cj01+X+4x6L1H5TPuqUkyZ6gD04ynUhbko4dhlanALcriF7t
  IfQR9i2r2x0yv5gxJEW/zztGqlwma/d4rBoPjnf6t07rlFHXMt/DVTkAfn5wxxtLdwkn5FMyvThRmex3BDf0gujoI1y6c0WLe9Y5geNX0oj+M
  Xg/W0cXAtzSFocstV1PoVqy883hNoeQZ3mIGB3Q0rIUUm5d9MA2bMMt31m1g3Sin6EQ== azureuser@myVM"
}
```

Execute the VMAccess script with:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name VMAccessForLinux \
--publisher Microsoft.OSTCExtensions \
--version 1.4 \
--protected-settings update_ssh_key.json
```

To reset a user password, create a file named `reset_user_password.json` and add settings in the following format. Replace `username` and `password` with your own information:

```
{
  "username": "azureuser",
  "password": "myNewPassword"
}
```

Execute the VMAccess script with:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name VMAccessForLinux \
--publisher Microsoft.OSTCExtensions \
--version 1.4 \
--protected-settings reset_user_password.json
```

## Restart SSH

To restart the SSH daemon and reset the SSH configuration to default values, create a file named `reset_sshd.json`. Add the following text:

```
{  
  "reset_ssh": true  
}
```

Execute the VMAccess script with:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name VMAccessForLinux \
--publisher Microsoft.OSTCExtensions \
--version 1.4 \
--protected-settings reset_sshd.json
```

## Manage administrative users

To create a user with `sudo` permissions that uses an SSH key for authentication, create a file named `create_new_user.json` and add settings in the following format. Substitute your own values for the `username` and `ssh_key` parameters. This method is designed to help you regain access to a VM in the event that current credentials are lost or forgotten. As a best practice, accounts with `sudo` permissions should be limited.

```
{  
  "username": "myNewUser",  
  "ssh_key": "ssh-rsa  
AAAAAB3NzaC1yc2EAAAQABAAQACQZ3S7gGp3rcbKmG2Y4vGZFMuMZCwoUzZNG1vHY7P2XV2x9FfAhy8iGD+1F8UdjFX3t5ebMm6BnnMh8  
fHwkTRd0t3LDQq8o8ElTBrZaKPxZN2tMZh0Ds5H1emb2UX0oRIGRcvWqsd4oJmxsXa/Si98Wa6RHwbc9QZhw80KAcoVhmndZAZAGR+Hq6ys  
lN05TMOr1/ZyQAook5C4FtcSGn3Y+WczaogWIxG4Zawk128g79VIeJcIQq0jPodHvQAh117qD1ItVvBfM0ben3GyhYTm7k4Yw1EdkONm4yV/  
UIW0la1rmyztSBQIm9sZmSq44XgjVmDHNF8UFcz1ToE4r2SdwTmZv00T2i5faeYnHzxiLPA3Enub7iUo5IdwFArnqad7M01SY1kLemhX9eF  
jLWN4mJe56Fu4NiWJkr9APSZQrYeKaqr4KUC68QpVasNjhbxPSf/PcjF3cj01+X+4x6L1H5HTPuqUkyZggD04ynUhbko4dhlanALcriF7t  
IfQR9i2r2x0yv5gxJEW/zztGqWma/d4rBoPjnf6t07rLFHXMt/DVTkAfn5woYtLDwkn5FMyvThRmex3BDf0gujoI1y6c0WLe9Y5geNX0oj+M  
Xg/W0cXAtzSFocstV1PoVqy883hNoeQZ3mIGB3Q0rIUm5d9MA2bMMt31m1g3Sin6EQ== myNewUser@myVM",  
  "password": "myNewUserPassword"  
}
```

Execute the VMAccess script with:

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myVM \
--name VMAccessForLinux \
--publisher Microsoft.OSTCExtensions \
--version 1.4 \
--protected-settings create_new_user.json
```

To delete a user, create a file named `delete_user.json` and add the following content. Change the data for

`remove_user` to the user you're trying to delete:

```
{  
  "remove_user": "myNewUser"  
}
```

Execute the VMAccess script with:

```
az vm extension set \  
  --resource-group myResourceGroup \  
  --vm-name myVM \  
  --name VMAccessForLinux \  
  --publisher Microsoft.OSTCExtensions \  
  --version 1.4 \  
  --protected-settings delete_user.json
```

## Check or repair the disk

Using VMAccess you can also check and repair a disk that you added to the Linux VM.

To check and then repair the disk, create a file named `disk_check_repair.json` and add settings in the following format. Change the data for `repair_disk` to the disk you're trying to repair:

```
{  
  "check_disk": "true",  
  "repair_disk": "true, mydiskname"  
}
```

Execute the VMAccess script with:

```
az vm extension set \  
  --resource-group myResourceGroup \  
  --vm-name myVM \  
  --name VMAccessForLinux \  
  --publisher Microsoft.OSTCExtensions \  
  --version 1.4 \  
  --protected-settings disk_check_repair.json
```

## Troubleshoot and support

### Troubleshoot

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myVM -o table
```

### Support

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Chef VM Extension for Linux and Windows

9/21/2022 • 3 minutes to read • [Edit Online](#)

Chef Software provides a DevOps automation platform for Linux and Windows that enables the management of both physical and virtual server configurations. The Chef VM Extension is an extension that enables Chef on virtual machines.

## Prerequisites

### Operating system

The Chef VM Extension is supported on all the [Extension Supported OS's](#) in Azure.

### Internet connectivity

The Chef VM Extension requires that the target virtual machine is connected to the internet in order to retrieve the Chef Client payload from the content delivery network (CDN).

## Extension schema

The following JSON shows the schema for the Chef VM Extension. The extension requires at a minimum the Chef Server URL, the Validation Client Name and the Validation Key for the Chef Server; these values can be found in the `knife.rb` file in the starter-kit.zip that is downloaded when you install [Chef Automate](#) or a standalone [Chef Server](#). Because the validation key should be treated as sensitive data, it should be configured under the `protectedSettings` element, meaning that it will only be decrypted on the target virtual machine.

```
{
  "type": "Microsoft.Compute/virtualMachines/extensions",
  "name": "[concat(variables('vmName'), '/', parameters('chef_vm_extension_type'))]",
  "apiVersion": "2017-12-01",
  "location": "[parameters('location')]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Chef.Bootstrap.WindowsAzure",
    "type": "[parameters('chef_vm_extension_type')]",
    "typeHandlerVersion": "1210.13",
    "settings": {
      "bootstrap_options": {
        "chef_server_url": "[parameters('chef_server_url')]",
        "validation_client_name": "[parameters('chef_validation_client_name')]"
      },
      "runlist": "[parameters('chef_runlist')]"
    },
    "protectedSettings": {
      "validation_key": "[parameters('chef_validation_key')]"
    }
  }
}
```

### Core property values

NAME	VALUE / EXAMPLE	DATA TYPE
apiVersion	2017-12-01	string (date)

NAME	VALUE / EXAMPLE	DATA TYPE
publisher	Chef.Bootstrap.WindowsAzure	string
type	LinuxChefClient (Linux), ChefClient (Windows)	string
typeHandlerVersion	1210.13	string (double)

## Settings

NAME	VALUE / EXAMPLE	DATA TYPE	REQUIRED?
settings/bootstrap_options/chef_server_url	https://api.chef.io/organizations/myorg	string (myorg)	Y
settings/bootstrap_options/validation_client_name	myorg-validator	string	Y
settings/runlist	recipe[mycookbook::default]	string	Y

## Protected settings

NAME	EXAMPLE	DATA TYPE	REQUIRED?
protectedSettings/validation_key	-----BEGIN RSA PRIVATE KEY-----\nKEYDATA\n-----END RSA PRIVATE KEY-----	string	Y

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. Templates can be used to deploy one or more virtual machines, install the Chef Client, connect to the Chef Server and the perform the initial configuration on the server as defined by the [Run-list](#)

A sample Resource Manager template that includes the Chef VM Extension can be found in the [Azure quickstart gallery](#).

The JSON configuration for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON configuration affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

## Azure CLI deployment

The Azure CLI can be used to deploy the Chef VM Extension to an existing VM. Replace the `validation_key` with the contents of your validation key (this file as a `.pem` extension). Replace `validation_client_name`, `chef_server_url` and `run_list` with those values from the `knife.rb` file in your Starter Kit.

```
az vm extension set \
--resource-group myResourceGroup \
--vm-name myExistingVM \
--name LinuxChefClient \
--publisher Chef.Bootstrap.WindowsAzure \
--version 1210.13 --protected-settings '{"validation_key": "<validation_key>"}' \
--settings '{ "bootstrap_options": { "chef_server_url": "<chef_server_url>", "validation_client_name": "<validation_client_name>" }, "runlist": "<run_list>" }'
```

## Troubleshooting and support

Data about the state of extension deployments can be retrieved from the Azure portal, and by using the Azure CLI. To see the deployment state of extensions for a given VM, run the following command using the Azure CLI.

```
az vm extension list --resource-group myResourceGroup --vm-name myExistingVM -o table
```

Extension execution output is logged to the following file:

### Linux

```
/var/lib/waagent/Chef.Bootstrap.WindowsAzure.LinuxChefClient
```

### Windows

```
C:\Packages\Plugins\Chef.Bootstrap.WindowsAzure.ChefClient\
```

### Error codes and their meanings

ERROR CODE	MEANING	POSSIBLE ACTION
51	This extension is not supported on the VM's operating system	

Additional troubleshooting information can be found in the [Chef VM Extension readme](#).

#### NOTE

For anything else directly related to Chef, contact [Chef Support](#).

## Next steps

If you need more help at any point in this article, you can contact the Azure experts on the [MSDN Azure and Stack Overflow forums](#). Alternatively, you can file an Azure support incident. Go to the [Azure support site](#) and select Get support. For information about using Azure Support, read the [Microsoft Azure support FAQ](#).

# Stackify Retrace Linux Agent Extension

9/21/2022 • 3 minutes to read • [Edit Online](#)

## Overview

Stackify provides products that track details about your application to help find and fix problems quickly. For developer teams, Retrace is a fully integrated, multi-environment, app performance super-power. It combines several tools every development team needs.

Retrace is the ONLY tool that delivers all of the following capabilities across all environments in a single platform.

- Application performance management (APM)
- Application and server logging
- Error tracking and monitoring
- Server, application, and custom metrics

### About Stackify Linux Agent Extension

This extension provides an install path for the Linux Agent for Retrace.

## Prerequisites

### Operating system

The Retrace agent can be run against these Linux distributions

DISTRIBUTION	VERSION
Ubuntu	16.04 LTS, 14.04 LTS, 16.10 and 17.04
Debian	7.9+ and 8.2+, 9
Red Hat	6.7+, 7.1+
CentOS	6.3+, 7.0+

### Internet connectivity

The Stackify Agent extension for Linux requires that the target virtual machine is connected to the internet.

You may need to adjust your network configuration to allow connections to Stackify, see <https://support.stackify.com/hc/en-us/articles/207891903-Adding-Exceptions-to-a-Firewall>.

## Extension schema

The following JSON shows the schema for the Stackify Retrace Agent extension. The extension requires the `environment` and `activationKey`.

```
{
  "type": "extensions",
  "name": "StackifyExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[resourceId('Microsoft.Compute/virtualMachines',variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Stackify.LinuxAgent.Extension",
    "type": "StackifyLinuxAgentExtension",
    "typeHandlerVersion": "1.0",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "environment": "myEnvironment"
    },
    "protectedSettings": {
      "activationKey": "myActivationKey"
    }
  }
}
```

## Template deployment

Azure VM extensions can be deployed with Azure Resource Manager templates. The JSON schema detailed in the previous section can be used in an Azure Resource Manager template to run the Stackify Retrace Linux Agent extension during an Azure Resource Manager template deployment.

The JSON for a virtual machine extension can be nested inside the virtual machine resource, or placed at the root or top level of a Resource Manager JSON template. The placement of the JSON affects the value of the resource name and type. For more information, see [Set name and type for child resources](#).

The following example assumes the Stackify Retrace Linux extension is nested inside the virtual machine resource. When nesting the extension resource, the JSON is placed in the "resources": [] object of the virtual machine.

The extension requires the `environment` and `activationKey`.

```
{
  "type": "extensions",
  "name": "StackifyExtension",
  "apiVersion": "[variables('apiVersion')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[resourceId('Microsoft.Compute/virtualMachines',variables('vmName'))]"
  ],
  "properties": {
    "publisher": "Stackify.LinuxAgent.Extension",
    "type": "StackifyLinuxAgentExtension",
    "typeHandlerVersion": "1.0",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "environment": "myEnvironment"
    },
    "protectedSettings": {
      "activationKey": "myActivationKey"
    }
  }
}
```

When placing the extension JSON at the root of the template, the resource name includes a reference to the

parent virtual machine, and the type reflects the nested configuration.

```
{  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "name": "<parentVmResource>/StackifyExtension",  
    "apiVersion": "[variables('apiVersion')]",  
    "location": "[resourceGroup().location]",  
    "dependsOn": [  
        "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"  
    ],  
    "properties": {  
        "publisher": "Stackify.LinuxAgent.Extension",  
        "type": "StackifyLinuxAgentExtension",  
        "typeHandlerVersion": "1.0",  
        "autoUpgradeMinorVersion": true,  
        "settings": {  
            "environment": "myEnvironment"  
        },  
        "protectedSettings": {  
            "activationKey": "myActivationKey"  
        }  
    }  
}
```

## PowerShell deployment

The `Set-AzVMExtension` command can be used to deploy the Stackify Retrace Linux Agent virtual machine extension to an existing virtual machine. Before running the command, the public and private configurations need to be stored in a PowerShell hash table.

The extension requires the `environment` and `activationKey`.

```
$PublicSettings = @{"environment" = "myEnvironment"}  
$ProtectedSettings = @{"activationKey" = "myActivationKey"}  
  
Set-AzVMExtension -ExtensionName "Stackify.LinuxAgent.Extension" `  
    -ResourceGroupName "myResourceGroup" `  
    -VMName "myVM" `  
    -Publisher "Stackify.LinuxAgent.Extension" `  
    -ExtensionType "StackifyLinuxAgentExtension" `  
    -TypeHandlerVersion 1.0 `  
    -Settings $PublicSettings `  
    -ProtectedSettings $ProtectedSettings `  
    -Location WestUS `
```

## Azure CLI deployment

The Azure CLI tool can be used to deploy the Stackify Retrace Linux Agent virtual machine extension to an existing virtual machine.

The extension requires the `environment` and `activationKey`.

```
az vm extension set --publisher 'Stackify.LinuxAgent.Extension' --version 1.0 --name  
'StackifyLinuxAgentExtension' --protected-settings '{"activationKey":"myActivationKey"}' --settings  
'{"environment":"myEnvironment"}' --resource-group 'myResourceGroup' --vm-name 'myVmName'
```

## Troubleshoot and support

## Error codes

ERROR CODE	MEANING	POSSIBLE ACTION
10	Install Error	wget is required
20	Install Error	Python is required
30	Install Error	sudo is required
40	Install Error	activationKey is required
51	Install Error	OS distro not supported
60	Install Error	environment is required
70	Install Error	Unknown
80	Enable Error	Service setup failed
90	Enable Error	Service startup failed
100	Disable Error	Service Stop Failed
110	Disable Error	Service Removal Failed
120	Uninstall Error	Service Stop Failed

If you need more help you can contact Stackify support at <https://support.stackify.com>.

# How to install and configure Symantec Endpoint Protection on a Windows VM

9/21/2022 • 2 minutes to read • [Edit Online](#)

## IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

Azure has two different deployment models for creating and working with resources: [Resource Manager and Classic](#). This article covers using the Classic deployment model. Microsoft recommends that most new deployments use the Resource Manager model.

This article shows you how to install and configure the Symantec Endpoint Protection client on an existing virtual machine (VM) running Windows Server. This full client includes services such as virus and spyware protection, firewall, and intrusion prevention. The client is installed as a security extension by using the VM Agent.

If you have an existing subscription from Symantec for an on-premises solution, you can use it to protect your Azure virtual machines. If you're not a customer yet, you can sign up for a trial subscription. For more information about this solution, see [Symantec Endpoint Protection on Microsoft's Azure platform](#). This page also has links to licensing information and instructions for installing the client if you're already a Symantec customer.

## Install Symantec Endpoint Protection on an existing VM

Before you begin, you need the following:

- The Azure PowerShell module, version 0.8.2 or later, on your work computer. You can check the version of Azure PowerShell that you have installed with the `Get-Module azure | format-table version` command. For instructions and a link to the latest version, see [How to Install and Configure Azure PowerShell](#). Log in to your Azure subscription using `Add-AzureAccount`.
- The VM Agent running on the Azure Virtual Machine.

First, verify that the VM Agent is already installed on the virtual machine. Fill in the cloud service name and virtual machine name, and then run the following commands at an administrator-level Azure PowerShell command prompt. Replace everything within the quotes, including the < and > characters.

## TIP

If you don't know the cloud service and virtual machine names, run `Get-AzureVM` to list the names for all virtual machines in your current subscription.

```
$CSName = "<cloud service name>"  
$VMName = "<virtual machine name>"  
$vm = Get-AzureVM -ServiceName $CSName -Name $VMName  
write-host $vm.VM.ProvisionGuestAgent
```

If the **write-host** command displays **True**, the VM Agent is installed. If it displays **False**, see the instructions and a link to the download in the Azure blog post [VM Agent and Extensions - Part 2](#).

If the VM Agent is installed, run these commands to install the Symantec Endpoint Protection agent.

```
$Agent = Get-AzureVMAvailableExtension -Publisher Symantec -ExtensionName SymantecEndpointProtection  
  
Set-AzureVMExtension -Publisher Symantec -Version $Agent.Version -ExtensionName SymantecEndpointProtection \  
-VM $vm | Update-AzureVM
```

To verify that the Symantec security extension has been installed and is up-to-date:

1. Log on to the virtual machine. For instructions, see [How to Log on to a Virtual Machine Running Windows Server](#).
2. For Windows Server 2008 R2, click **Start > Symantec Endpoint Protection**. For Windows Server 2012 or Windows Server 2012 R2, from the start screen, type **Symantec**, and then click **Symantec Endpoint Protection**.
3. From the **Status** tab of the **Status-Symantec Endpoint Protection** window, apply updates or restart if needed.

## Additional resources

[How to Log on to a Virtual Machine Running Windows Server](#)

[Azure VM Extensions and Features](#)

# Use Azure Policy to restrict extensions installation on Linux VMs

9/21/2022 • 3 minutes to read • [Edit Online](#)

If you want to prevent the installation of certain extensions on your Linux VMs, you can create an Azure Policy definition using the Azure CLI to restrict extensions for VMs within a resource group. To learn the basics of Azure VM extensions for Linux, see [Virtual machine extensions and features for Linux](#).

This tutorial uses the CLI within the Azure Cloud Shell, which is constantly updated to the latest version. If you want to run the Azure CLI locally, you need to install version 2.0.26 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create a rules file

In order to restrict what extensions are available, you need to create a [rule](#) to identify the extension.

This example demonstrates how to deny the installation of disallowed VM extensions by defining a rules file in Azure Cloud Shell. However, if you're working in Azure CLI locally, you can create a local file and replace the path (~/.clouddrive) with the path to the file on your local file system.

In a [bash Cloud Shell](#), type:

```
vim ~/.clouddrive/azurepolicy.rules.json
```

Copy and paste the following `.json` data into the file.

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Compute/virtualMachines/extensions"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/publisher",
        "equals": "Microsoft.OSTCExtensions"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/type",
        "in": "[parameters('notAllowedExtensions')]"
      }
    ]
  },
  "then": {
    "effect": "deny"
  }
}
```

When you're finished, press `Esc`, and then type `:wq` to save and close the file.

## Create a parameters file

You also need a [parameters](#) file that creates a structure for you to use for passing in a list of the unauthorized

extensions.

This example shows you how to create a parameter file for Linux VMs in Cloud Shell.

In the bash Cloud Shell opened before type:

```
vim ~/clouddrive/azurepolicy.parameters.json
```

Copy and paste the following `.json` data into the file.

```
{
  "notAllowedExtensions": {
    "type": "Array",
    "metadata": {
      "description": "The list of extensions that will be denied. Example: CustomScriptForLinux, VMAccessForLinux etc.",
      "displayName": "Denied extension"
    }
  }
}
```

When you're finished, press `Esc`, and then type `:wq` to save and close the file.

## Create the policy

A *policy definition* is an object used to store the configuration that you would like to use. The policy definition uses the rules and parameters files to define the policy. Create the policy definition using [az policy definition create](#).

In this example, the rules and parameters are the files you created and stored as `.json` files in Cloud Shell or in your local file system.

```
az policy definition create \
  --name 'not-allowed-vmextension-linux' \
  --display-name 'Block VM Extensions' \
  --description 'This policy governs which VM extensions that are blocked.' \
  --rules '~/clouddrive/azurepolicy.rules.json' \
  --params '~/clouddrive/azurepolicy.parameters.json' \
  --mode All
```

## Assign the policy

This example assigns the policy to a resource group using `az policy assignment create`. Any VM created in the `myResourceGroup` resource group will be unable to install the Linux VM Access or the Custom Script Extensions for Linux.

### NOTE

The resource group must exist before you can assign the policy.

Use `az account list` to find your subscription ID and replace the placeholder in the following example:

```
az policy assignment create \
    --name 'not-allowed-vmextension-linux' \
    --scope /subscriptions/<subscription Id>/resourceGroups/myResourceGroup \
    --policy "not-allowed-vmextension-linux" \
    --params '{
    "notAllowedExtensions": {
        "value": [
            "VMAccessForLinux",
            "CustomScriptForLinux"
        ]
    }
}'
```

## Test the policy

Test the policy by creating a new VM and adding a new user.

```
az vm create \
    --resource-group myResourceGroup \
    --name myVM \
    --image UbuntuLTS \
    --generate-ssh-keys
```

Try to create a new user named **myNewUser** using the VM Access extension.

```
az vm user update \
    --resource-group myResourceGroup \
    --name myVM \
    --username myNewUser \
    --password 'mynewuserpwd123!'
```

## Remove the assignment

```
az policy assignment delete --name 'not-allowed-vmextension-linux' --resource-group myResourceGroup
```

## Remove the policy

```
az policy definition delete --name 'not-allowed-vmextension-linux'
```

## Next steps

For more information, see [Azure Policy](#).

# Use Azure Policy to restrict extensions installation on Windows VMs

9/21/2022 • 2 minutes to read • [Edit Online](#)

If you want to prevent the use or installation of certain extensions on your Windows VMs, you can create an Azure Policy definition using PowerShell to restrict extensions for VMs within a resource group.

This tutorial uses Azure PowerShell within the Cloud Shell, which is constantly updated to the latest version.

## Create a rules file

In order to restrict what extensions can be installed, you need to have a [rule](#) to provide the logic to identify the extension.

This example shows you how to deny extensions published by 'Microsoft.Compute' by creating a rules file in Azure Cloud Shell, but if you are working in PowerShell locally, you can also create a local file and replace the path (\$home/clouddrive) with the path to the local file on your machine.

In a [Cloud Shell](#), type:

```
nano $home/clouddrive/rules.json
```

Copy and paste the following json into the file.

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Compute/virtualMachines/extensions"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/publisher",
        "equals": "Microsoft.Compute"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/type",
        "in": "[parameters('notAllowedExtensions')]"
      }
    ]
  },
  "then": {
    "effect": "deny"
  }
}
```

When you are done, hit the **Ctrl + O** and then **Enter** to save the file. Hit **Ctrl + X** to close the file and exit.

## Create a parameters file

You also need a [parameters](#) file that creates a structure for you to use for passing in a list of the extensions to block.

This example shows you how to create a parameters file for VMs in Cloud Shell, but if you are working in

PowerShell locally, you can also create a local file and replace the path (\$home/clouddrive) with the path to the local file on your machine.

In [Cloud Shell](#), type:

```
nano $home/clouddrive/parameters.json
```

Copy and paste the following json into the file.

```
{
  "notAllowedExtensions": {
    "type": "Array",
    "metadata": {
      "description": "The list of extensions that will be denied.",
      "displayName": "Denied extension"
    }
  }
}
```

When you are done, hit the **Ctrl + O** and then **Enter** to save the file. Hit **Ctrl + X** to close the file and exit.

## Create the policy

A policy definition is an object used to store the configuration that you would like to use. The policy definition uses the rules and parameters files to define the policy. Create a policy definition using the [New-AzPolicyDefinition](#) cmdlet.

The policy rules and parameters are the files you created and stored as .json files in your cloud shell.

```
$definition = New-AzPolicyDefinition ` 
  -Name "not-allowed-vmextension-windows" ` 
  -DisplayName "Not allowed VM Extensions" ` 
  -description "This policy governs which VM extensions that are explicitly denied." ` 
  -Policy 'C:\Users\ContainerAdministrator\clouddrive\rules.json' ` 
  -Parameter 'C:\Users\ContainerAdministrator\clouddrive\parameters.json'
```

## Assign the policy

This example assigns the policy to a resource group using [New-AzPolicyAssignment](#). Any VM created in the **myResourceGroup** resource group will not be able to install the VM Access Agent or Custom Script extensions.

Use the [Get-AzSubscription | Format-Table](#) cmdlet to get your subscription ID to use in place of the one in the example.

```
$scope = "/subscriptions/<subscription id>/resourceGroups/myResourceGroup"
$assignment = New-AzPolicyAssignment ` 
    -Name "not-allowed-vmextension-windows" ` 
    -Scope $scope ` 
    -PolicyDefinition $definition ` 
    -PolicyParameter '{
        "notAllowedExtensions": {
            "value": [
                "VMAccessAgent",
                "CustomScriptExtension"
            ]
        }
    }'
$assignment
```

## Test the policy

To test the policy, try to use the VM Access extension. The following should fail with the message "Set-AzVMAccessExtension : Resource 'myVMAccess' was disallowed by policy."

```
Set-AzVMAccessExtension ` 
    -ResourceGroupName "myResourceGroup" ` 
    -VMName "myVM" ` 
    -Name "myVMAccess" ` 
    -Location EastUS
```

In the portal, the password change should fail with the "The template deployment failed because of policy violation." message.

## Remove the assignment

```
Remove-AzPolicyAssignment -Name not-allowed-vmextension-windows -Scope $scope
```

## Remove the policy

```
Remove-AzPolicyDefinition -Name not-allowed-vmextension-windows
```

## Next steps

For more information, see [Azure Policy](#).

# How to update the Azure Linux Agent on a VM

9/21/2022 • 5 minutes to read • [Edit Online](#)

To update your [Azure Linux Agent](#) on a Linux VM in Azure, you must already have:

- A running Linux VM in Azure.
- A connection to that Linux VM using SSH.

You should always check for a package in the Linux distro repository first. It is possible the package available may not be the latest version, however, enabling autoupdate will ensure the Linux Agent will always get the latest update. Should you have issues installing from the package managers, you should seek support from the distro vendor.

## NOTE

For more information see [Endorsed Linux distributions on Azure](#)

Verify the [Minimum version support for virtual machine agents in Azure](#) before proceeding.

## Ubuntu

Check your current package version

```
apt list --installed | grep walinuxagent
```

Update package cache

```
sudo apt-get -qq update
```

Install the latest package version

```
sudo apt-get install walinuxagent
```

Ensure auto update is enabled. First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=y
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/# AutoUpdate.Enabled=n/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

Restart waagent service for 14.04

```
initctl restart walinuxagent
```

Restart waagent service for 16.04 / 17.04

```
systemctl restart walinuxagent.service
```

## Red Hat / CentOS

### RHEL/CentOS 6

Check your current package version

```
sudo yum list WALinuxAgent
```

Check available updates

```
sudo yum check-update WALinuxAgent
```

Install the latest package version

```
sudo yum install WALinuxAgent
```

Ensure auto update is enabled

First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=y
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/\# AutoUpdate.Enabled=y/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

Restart the waagent service

```
sudo service waagent restart
```

## RHEL/CentOS 7

Check your current package version

```
sudo yum list WALinuxAgent
```

Check available updates

```
sudo yum check-update WALinuxAgent
```

Install the latest package version

```
sudo yum install WALinuxAgent
```

Ensure auto update is enabled. First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=y  
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/# AutoUpdate.Enabled=n/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

Restart the waagent service

```
sudo systemctl restart waagent.service
```

## SUSE SLES

### SUSE SLES 11 SP4

Check your current package version

```
zypper info python-azure-agent
```

Check available updates. The above output will show you if the package is up to date.

Install the latest package version

```
sudo zypper install python-azure-agent
```

Ensure auto update is enabled

First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=y  
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/# AutoUpdate.Enabled=n/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

Restart the waagent service

```
sudo /etc/init.d/waagent restart
```

## SUSE SLES 12 SP2

Check your current package version

```
zypper info python-azure-agent
```

Check available updates

In the output from the above, this will show you if the package is up-to-date.

Install the latest package version

```
sudo zypper install python-azure-agent
```

Ensure auto update is enabled

First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/AutoUpdate.Enabled=n.*/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

Restart the waagent service

```
sudo systemctl restart waagent.service
```

## Debian

### Debian 7 "Jesse"/ Debian 7 "Stretch"

Check your current package version

```
dpkg -l | grep waagent
```

Update package cache

```
sudo apt-get -qq update
```

Install the latest package version

```
sudo apt-get install waagent
```

Enable agent auto update This version of Debian does not have a version >= 2.0.16, therefore AutoUpdate is not available for it. The output from the above command will show you if the package is up-to-date.

### **Debian 8 “Jessie” / Debian 9 “Stretch”**

Check your current package version

```
apt list --installed | grep waagent
```

Update package cache

```
sudo apt-get -qq update
```

Install the latest package version

```
sudo apt-get install waagent
```

Ensure auto update is enabled First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
AutoUpdate.Enabled=y  
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/AutoUpdate.Enabled=n.*/AutoUpdate.Enabled=y/g' /etc/waagent.conf  
Restart the waagent service  
sudo systemctl restart walinuxagent.service
```

## Oracle Linux 6 and Oracle Linux 7

For Oracle Linux, make sure that the `Addons` repository is enabled. Choose to edit the file

`/etc/yum.repos.d/public-yum-ol6.repo` (Oracle Linux 6) or `/etc/yum.repos.d/oracle-linux-ol7.repo` (Oracle Linux), and change the line `enabled=0` to `enabled=1` under `[ol6_addons]` or `[ol7_addons]` in this file.

Then, to install the latest version of the Azure Linux Agent, type:

```
sudo yum install WALinuxAgent
```

If you don't find the add-on repository you can simply add these lines at the end of your .repo file according to your Oracle Linux release:

For Oracle Linux 6 virtual machines:

```
[ol6_addons]
name=Add-Ons for Oracle Linux $releasever ($basearch)
baseurl=https://public-yum.oracle.com/repo/OracleLinux/OL6/addons/x86_64
gpgkey=https://public-yum.oracle.com/RPM-GPG-KEY-oracle-ol6
gpgcheck=1
enabled=1
```

For Oracle Linux 7 virtual machines:

```
[ol7_addons]
name=Oracle Linux $releasever Add ons ($basearch)
baseurl=http://public-yum.oracle.com/repo/OracleLinux/OL7/addons/$basearch/
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
gpgcheck=1
enabled=1
```

Then type:

```
sudo yum update WALinuxAgent
```

Typically this is all you need, but if for some reason you need to install it from <https://github.com> directly, use the following steps.

## Update the Linux Agent when no agent package exists for distribution

Install wget (there are some distros that don't install it by default, such as Red Hat, CentOS, and Oracle Linux versions 6.4 and 6.5) by typing `sudo yum install wget` on the command line.

### 1. Download the latest version

Open [the release of Azure Linux Agent in GitHub](https://github.com/Azure/WALinuxAgent/releases) in a web page, and find out the latest version number. (You can locate your current version by typing `waagent --version`.)

For version 2.2.x or later, type:

```
wget https://github.com/Azure/WALinuxAgent/archive/refs/tags/v2.2.x.zip
unzip v2.2.x.zip
cd WALinuxAgent-2.2.x
```

The following line uses version 2.2.14 as an example:

```
wget https://github.com/Azure/WALinuxAgent/archive/refs/tags/v2.2.14.zip
unzip v2.2.14.zip
cd WALinuxAgent-2.2.14
```

### 2. Install the Azure Linux Agent

For version 2.2.x, use: You may need to install the package `setuptools` first--see [setuptools](#). Then run:

```
sudo python setup.py install
```

Ensure auto update is enabled. First, check to see if it is enabled:

```
cat /etc/waagent.conf
```

Find 'AutoUpdate.Enabled'. If you see this output, it is enabled:

```
# AutoUpdate.Enabled=y  
AutoUpdate.Enabled=y
```

To enable run:

```
sudo sed -i 's/# AutoUpdate.Enabled=n/AutoUpdate.Enabled=y/g' /etc/waagent.conf
```

### 3. Restart the waagent service

For most of Linux distros:

```
sudo service waagent restart
```

For Ubuntu, use:

```
sudo service walinuxagent restart
```

For CoreOS, use:

```
sudo systemctl restart waagent
```

### 4. Confirm the Azure Linux Agent version

```
waagent -version
```

For CoreOS, the above command may not work.

You will see that the Azure Linux Agent version has been updated to the new version.

For more information regarding the Azure Linux Agent, see [Azure Linux Agent README](#).

# Exporting Resource Groups that contain VM extensions

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure Resource Groups can be exported into a new Resource Manager template that can then be redeployed. The export process interprets existing resources, and creates a Resource Manager template that when deployed results in a similar Resource Group. When using the Resource Group export option against a Resource Group containing Virtual Machine extensions, several items need to be considered such as extension compatibility and protected settings.

This document details how the Resource Group export process works regarding virtual machine extensions, including a list of supported extensions, and details on handling secured data.

## Supported Virtual Machine Extensions

Many Virtual Machine extensions are available. Not all extensions can be exported into a Resource Manager template using the "Automation Script" feature. If a virtual machine extension is not supported, it needs to be manually placed back into the exported template.

The following extensions can be exported with the automation script feature.

Acronis Backup, Acronis Backup Linux, Bg Info, BMC CTM Agent Linux, BMC CTM Agent Windows, Chef Client, Custom Script, Custom Script Extension, Custom Script for Linux, Datadog Linux Agent, Datadog Windows Agent, Docker Extension, DSC Extension, Dynatrace Linux, Dynatrace Windows, HPE Security Application Defender for Cloud, IaaS Antimalware, IaaS Diagnostics, Linux Chef Client, Linux Diagnostic, OS Patching For Linux, Puppet Agent, Site 24x7 Apm Insight, Site 24x7 Linux Server, Site 24x7 Windows Server, Trend Micro DSA, Trend Micro DSA Linux, VM Access For Linux, VM Access For Linux, VM Snapshot, VM Snapshot Linux

## Export the Resource Group

To export a Resource Group into a reusable template, complete the following steps:

1. Sign in to the Azure portal
2. On the Hub Menu, click Resource Groups
3. Select the target resource group from the list
4. In the Resource Group blade, click Automation Script

The screenshot shows the Azure Resource Manager template export interface. At the top, there are buttons for Download, Add to library, and Deploy. Below that is a header bar with an info icon and the text: "Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Define resources and configurable input parameters and deploy with script or code. [Learn more about template deployment](#)". The main area has tabs for Template, Parameters, CLI, PowerShell, .NET, and Ruby. The Template tab is selected, showing a tree view of resources: Parameters (8), Variables (0), and Resources (6). The Resources section contains items like [parameters('virtualMachines\_My...'), [parameters('networkInterfaces\_m...'), [parameters('publicIPAddresses\_m...'), [parameters('virtualNetworks\_MyV...'), [parameters('storageAccounts\_dia...'), and AzureDiagnostics (Microsoft.Com...]. Below the tree is a large block of JSON code:

```

1 {
2   "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "virtualMachines_MyWindowsVM_adminPassword": {
6       "defaultValue": null,
7       "type": "SecureString"
8     },
9     "extensions_Microsoft.Insights.VMDiagnosticsSettings_protectedSettings": {
10       "defaultValue": null,
11       "type": "SecureObject"
12     },
13     "virtualMachines_MyWindowsVM_name": {
14       "defaultValue": "MyWindowsVM",
15       "type": "String"
16     },
17     "networkInterfaces_myVMNic_name": {
18       "defaultValue": "myVMNic",
19       "type": "String"
20     },
21     "publicIPAddresses_myPublicIP_name": {
22       "defaultValue": "myPublicIP",
23       "type": "String"
24     },
25     "virtualNetworks_MyVNET_name": {
26       "defaultValue": "MyVNET",
27     }
}

```

The Azure Resource Manager automations script produces a Resource Manager template, a parameters file, and several sample deployment scripts such as PowerShell and Azure CLI. At this point, the exported template can be downloaded using the download button, added as a new template to the template library, or redeployed using the deploy button.

## Configure protected settings

Many Azure virtual machine extensions include a protected settings configuration, that encrypts sensitive data such as credentials and configuration strings. Protected settings are not exported with the automation script. If necessary, protected settings need to be reinserted into the exported template.

### Step 1 - Remove template parameter

When the Resource Group is exported, a single template parameter is created to provide a value to the exported protected settings. This parameter can be removed. To remove the parameter, look through the parameter list and delete the parameter that looks similar to this JSON example.

```
"extensions_extensionname_protectedSettings": {
  "defaultValue": null,
  "type": "SecureObject"
}
```

### Step 2 - Get protected settings properties

Because each protected setting has a set of required properties, a list of these properties need to be gathered. Each parameter of the protected settings configuration can be found in the [Azure Resource Manager schema on GitHub](#). This schema only includes the parameter sets for the extensions listed in the overview section of this document.

From within the schema repository, search for the desired extension, for this example `IaaSDiagnistics`. Once the extensions `protectedSettings` object has been located, take note of each parameter. In the example of the `IaaSDiagnostic` extension, the require parameters are `storageAccountName`, `storageAccountKey`, and `storageAccountEndPoint`.

```
"protectedSettings": {  
  "type": "object",  
  "properties": {  
    "storageAccountName": {  
      "type": "string"  
    },  
    "storageAccountKey": {  
      "type": "string"  
    },  
    "storageAccountEndPoint": {  
      "type": "string"  
    }  
  },  
  "required": [  
    "storageAccountName",  
    "storageAccountKey",  
    "storageAccountEndPoint"  
  ]  
}
```

### Step 3 - Re-create the protected configuration

On the exported template, search for `protectedSettings` and replace the exported protected setting object with a new one that includes the required extension parameters and a value for each one.

In the example of the `IaaSDiagnostic` extension, the new protected setting configuration would look like the following example:

```
"protectedSettings": {  
  "storageAccountName": "[parameters('storageAccountName')]",  
  "storageAccountKey": "[parameters('storageAccountKey')]",  
  "storageAccountEndPoint": "https://core.windows.net"  
}
```

The final extension resource looks similar to the following JSON example:

```
{
  "name": "Microsoft.Insights.VMDiagnosticsSettings",
  "type": "extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "[variables('apiVersion')]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/', variables('vmName'))]"
  ],
  "tags": {
    "displayName": "AzureDiagnostics"
  },
  "properties": {
    "publisher": "Microsoft.Azure.Diagnostics",
    "type": "IaaS.Diagnostics",
    "typeHandlerVersion": "1.5",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "xmlCfg": "[base64(concat(variables('wadcfgxstart'), variables('wadmetricsresourceid'),
variables('vmName'), variables('wadcfgxend')))]",
      "storageAccount": "[parameters('existingdiagnosticsStorageAccountName')]"
    },
    "protectedSettings": {
      "storageAccountName": "[parameters('storageAccountName')]",
      "storageAccountKey": "[parameters('storageAccountKey')]",
      "storageAccountEndPoint": "https://core.windows.net"
    }
  }
}
```

If using template parameters to provide property values, these need to be created. When creating template parameters for protected setting values, make sure to use the `SecureString` parameter type so that sensitive values are secured. For more information on using parameters, see [Authoring Azure Resource Manager templates](#).

In the example of the `IaaSDiagnostic` extension, the following parameters would be created in the parameters section of the Resource Manager template.

```
"storageAccountName": {
  "defaultValue": null,
  "type": "SecureString"
},
"storageAccountKey": {
  "defaultValue": null,
  "type": "SecureString"
}
```

At this point, the template can be deployed using any template deployment method.

# Troubleshooting Azure Windows VM extension failures

9/21/2022 • 7 minutes to read • [Edit Online](#)

## Overview of Azure Resource Manager templates

Azure Resource Manager templates allows you to declaratively specify the Azure IaaS infrastructure in JSON language by defining the dependencies between resources.

See [Authoring extension templates](#) to learn more about authoring templates for using extensions.

In this article we'll learn about troubleshooting some of the common VM extension failures.

## Viewing extension status

Azure Resource Manager templates can be executed from Azure PowerShell. Once the template is executed, the extension status can be viewed from Azure Resource Explorer or the command-line tools.

Here's an example:

Azure PowerShell:

```
Get-AzVM -ResourceGroupName $RGName -Name $vmName -Status
```

Here's the sample output:

```
Extensions: {
  "ExtensionType": "Microsoft.Compute.CustomScriptExtension",
  "Name": "myCustomScriptExtension",
  "SubStatuses": [
    {
      "Code": "ComponentStatus/StdOut/succeeded",
      "DisplayStatus": "Provisioning succeeded",
      "Level": "Info",
      "Message": "Directory: C:\\\\temp\\\\n\\\\n\\\\nMode
\\n---- ----- -----
9/1/2015 2:03 AM 11
test.txt \\n\\n",
      "Time": null
    },
    {
      "Code": "ComponentStatus/StdErr/succeeded",
      "DisplayStatus": "Provisioning succeeded",
      "Level": "Info",
      "Message": "",
      "Time": null
    }
  ]
}
```

## Troubleshooting extension failures

### Verify that the VM Agent is running and Ready

The VM Agent is required to manage, install and execute extensions. If the VM Agent isn't running or is failing to

report a Ready status to the Azure platform, then the extensions won't work correctly.

Please refer to the following pages to troubleshoot the VM Agent:

- [Troubleshooting Windows Azure Guest Agent](#) for a Windows VM
- [Troubleshoot the Azure Linux Agent](#) for a Linux VM

### Check for your specific extension troubleshooting guide

Some extensions have a specific page describing how to troubleshoot them. You can find the list of these extensions and pages on [Troubleshoot extensions](#).

### View the extension's status

As explained above, the extension's status can be found by running the PowerShell cmdlet:

```
Get-AzVM -ResourceGroupName $RGName -Name $vmName -Status
```

or the CLI command:

```
az vm extension show -g <RG Name> --vm-name <VM Name> --name <Extension Name>
```

or in the Azure portal, by browsing to the VM Blade / Settings / Extensions. You can then click on the extension and check its status and message.

### Rerun the extension on the VM

If you're running scripts on the VM using Custom Script Extension, you could sometimes run into an error where VM was created successfully but the script has failed. Under these conditions, the recommended way to recover from this error is to remove the extension and rerun the template again. Note: In future, this functionality would be enhanced to remove the need for uninstalling the extension.

#### Remove the extension from Azure PowerShell

```
Remove-AzVMExtension -ResourceGroupName $RGName -VMName $vmName -Name "myCustomScriptExtension"
```

Once the extension has been removed, the template can be re-executed to run the scripts on the VM.

### Trigger a new GoalState to the VM

You might notice that an extension hasn't been executed, or is failing to execute because of a missing "Windows Azure CRP Certificate Generator" (that certificate is used to secure the transport of the extension's protected settings). That certificate will be automatically regenerated by restarting the Windows Guest Agent from inside the Virtual Machine:

- Open the Task Manager
- Go to the Details tab
- Locate the WindowsAzureGuestAgent.exe process
- Right-click, and select "End Task". The process will be automatically restarted

You can also trigger a new GoalState to the VM, by executing a "VM Reapply". VM [Reapply](#) is an API introduced in 2020 to reapply a VM's state. We recommend doing this at a time when you can tolerate a short VM downtime. While Reapply itself doesn't cause a VM reboot, and the vast majority of times calling Reapply won't reboot the VM, there's a very small risk that some other pending update to the VM model gets applied when Reapply triggers a new goal state, and that other change could require a restart.

Azure portal:

In the portal, select the VM and in the left pane under the **Support + troubleshooting**, select **Redeploy +**

**reapply**, then select **Reapply**.

Azure PowerShell (*replace the RG Name and VM Name with your values*):

```
Set-AzVM -ResourceGroupName <RG Name> -Name <VM Name> -Reapply
```

Azure CLI (*replace the RG Name and VM Name with your values*):

```
az vm reapply -g <RG Name> -n <VM Name>
```

If a "VM Reapply" didn't work, you can add a new empty Data Disk to the VM from the Azure Management Portal, and then remove it later once the certificate has been added back.

### Look at the extension logs inside the VM

If the previous steps didn't work and if your extension is still in a failed state, the next step is to look at its logs inside the Virtual Machine.

On a **Windows** VM, the extension logs will typically reside in

```
C:\WindowsAzure\Logs\Plugins
```

And the Extension settings and status files will be in

```
C:\Packages\Plugins
```

On a **Linux** VM, the extension logs will typically reside in

```
/var/log/azure/
```

And the Extension settings and status files will be in

```
/var/lib/waagent/
```

Each extension is different, but they usually follow similar principles:

Extension packages and binaries are downloaded on the VM (eg. `"/var/lib/waagent/custom-script/download/1"` for Linux or `"C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.10.12\Downloads\0"` for Windows).

Their configuration and settings are passed from Azure Platform to the extension handler through the VM Agent (eg. `"/var/lib/waagent/Microsoft.Azure.Extensions.CustomScript-2.1.3/config"` for Linux or `"C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.10.12\RuntimeSettings"` for Windows)

Extension handlers inside the VM are writing to a status file (eg. `"/var/lib/waagent/Microsoft.Azure.Extensions.CustomScript-2.1.3/status/1.status"` for Linux or `"C:\Packages\Plugins\Microsoft.Compute.CustomScriptExtension\1.10.12>Status"` for Windows) which will then be reported to the Azure Platform. That status is the one reported through PowerShell, CLI or in the VM's extension blade in the Azure portal.

They also write detailed logs of their execution (eg. `"/var/log/azure/custom-script/handler.log"` for Linux or `"C:\WindowsAzure\Logs\Plugins\Microsoft.Compute.CustomScriptExtension\1.10.12\CustomScriptHandler.log"`

for Windows).

## If the VM is recreated from an existing VM

It could happen that you're creating an Azure VM based on a specialized Disk coming from another Azure VM. In that case, it's possible that the old VM contained extensions, and so will have binaries, logs and status files left over. The new VM model won't be aware of the previous VM's extensions states, and it might report an incorrect status for these extensions. We strongly recommend you to remove the extensions from the old VM before creating the new one, and then reinstall these extensions once the new VM is created. The same can happen when you create a generalized image from an existing Azure VM. We invite you to remove extensions to avoid inconsistent state from the extensions.

## Known issues

### PowerShell isn't recognized as an internal or external command

You notice the following error entries in the RunCommand extension's output:

```
RunCommandExtension failed with "'powershell' isn't recognized as an internal or external command,"
```

### Analysis

Extensions run under Local System account, so it's very possible that powershell.exe works fine when you RDP into the VM, but fails when run with RunCommand.

### Solution

- Check that PowerShell is properly listed in the PATH environment variable:
  - Open Control Panel
  - System and Security
  - System
  - Advanced tab -> Environmental Variables
- Under 'System variables' click edit and ensure that PowerShell is in the PATH environment variable (usually: "C:\Windows\System32\WindowsPowerShell\v1.0")
- Reboot the VM or restart the WindowsAzureGuestAgent service then try the Run Command again.

### Command isn't recognized as an internal or external command

You see the following in the C:\WindowsAzure\Logs\Plugins<ExtensionName><Version>\CommandExecution.log file:

```
Execution Error: '<command>' isn't recognized as an internal or external command, operable program or batch file.
```

### Analysis

Extensions run under Local System account, so it's very possible that powershell.exe works fine when you RDP into the VM, but fails when run with RunCommand.

### Solution

- Open a Command Prompt in the VM and execute a command to reproduce the error. The VM Agent uses the Administrator cmd.exe and you may have some preconfigured command to execute every time cmd is started.
- It's also likely that your PATH variable is misconfigured, but this will depend on the command that is having the problem.

**VMAccessAgent is failing with Cannot update Remote Desktop Connection settings for Administrator account. Error: System.Runtime.InteropServices.COMException (0x800706D9): There are no more endpoints available from the endpoint mapper.**

You see the following in the extension's status:

```
Type Microsoft.Compute.VMAccessAgent
Version 2.4.8
Status Provisioning failed
Status level Error
Status message Cannot update Remote Desktop Connection settings for Administrator account. Error:
System.Runtime.InteropServices.COMException (0x800706D9): There are no more endpoints available from the
endpoint mapper. (Exception from HRESULT: 0x800706D9) at NetFwTypeLib.INetFwRules.GetEnumerator() at
Microsoft.WindowsAzure.GuestAgent.Plugins.JsonExtensions.VMAccess.RemoteDesktopManager.EnableRemoteDesktopFi
rewallRules()
at
Microsoft.WindowsAzure.GuestAgent.Plugins.JsonExtensions.VMAccess.RemoteDesktopManager.EnableRemoteDesktop()
at
```

## Analysis

This error can happen when the Windows Firewall service isn't running.

## Solution

Check if the Windows Firewall service is enabled and running. If it's not, please enable and start it - then try again to run the VMAccessAgent.

## The remote certificate is invalid according to the validation procedure.

You see the following in the WaAppAgent.log

```
System.Net.WebException: The underlying connection was closed: Could not establish trust relationship for
the SSL/TLS secure channel. ---> System.Security.AuthenticationException: The remote certificate is invalid according to the validation
procedure.
```

## Analysis

Your VM is probably missing the Baltimore CyberTrust Root certificate in "Trusted Root Certification Authorities".

## Solution

Open the certificates console with certmgr.msc, and check if the certificate is there.

Another possible issue is that the certificate chain is broken by a third party SSL Inspection tool, like ZScaler. That kind of tool should be configured to bypass SSL inspection.

# Issues using VM extensions in Python 3-enabled Linux Azure Virtual Machines systems

9/21/2022 • 2 minutes to read • [Edit Online](#)

## NOTE

Microsoft encourages users to adopt **Python 3.x** in their systems unless your workload requires **Python 2.x** support. Examples of this requirement might include legacy administration scripts, or extensions such as **Azure Disk Encryption** and **Azure Monitor**.

Before installing **Python 2.x** in production, consider the question of long-term support of Python 2.x, particularly their ability to receive security updates. As products, including some of the extension mentioned, update with **Python 3.8** support, you should discontinue use of Python 2.x.

Some Linux distributions have transitioned to Python 3.8 and removed the legacy `/usr/bin/python` entrypoint for Python altogether. This transition impacts the out-of-the-box, automated deployment of certain virtual machine (VM) extensions with these two conditions:

- Extensions that are still transitioning to Python 3.x support
- Extensions that use the legacy `/usr/bin/python` entrypoint

Linux distribution users who have transitioned to **Python 3.x** must ensure the legacy `/usr/bin/python` entrypoint exists before attempting to deploy those extensions to their VMs. Otherwise, the extension deployment might fail.

- Endorsed Linux distributions that are affected include **Ubuntu Server 20.04 LTS** and **Ubuntu Pro 20.04 LTS**.
- Affected VM Extensions include **Azure Disk Encryption**, **Log Analytics**, **VM Access** (used for Password Reset), and **Guest Diagnostics** (used for additional performance counters).

In-place upgrades, such as upgrading from **Ubuntu 18.04 LTS** to **Ubuntu 20.04 LTS**, should retain the `/usr/bin/python` symlink, and remain unaffected.

## Resolution

Consider these general recommendations before deploying extensions in the known-affected scenarios described previously in the Summary:

1. Before deploying the extension, reinstate the `/usr/bin/python` symlink by using the Linux distribution vendor-provided method.
  - For example, for **Python 2.7**, use: `sudo apt update && sudo apt install python-is-python2`
2. This recommendation is for Azure customers and is not supported in Azure Stack:
  - If you've already deployed an instance that exhibits this problem, use the Run command functionality in the VM blade to run the commands mentioned above. The Run command extension itself is not affected by the transition to Python 3.8.
3. If you are deploying a new instance, and need to set an extension at provisioning time, use **cloud-init** user data to install the packages mentioned above.

For example, for Python 2.7:

```
# create cloud-init config
cat > cloudinitConfig.json <<EOF
#cloud-config
package_update: true

runcmd:
- sudo apt update
- sudo apt install python-is-python2
EOF

# create VM
az vm create \
--resource-group <resourceGroupName> \
--name <vmName> \
--image <Ubuntu 20.04 Image URN> \
--admin-username azadmin \
--ssh-key-value "<sshPubKey>" \
--custom-data ./cloudinitConfig.json
```

4. If your organization's policy administrators determine that extensions shouldn't be deployed in VMs, you can disable extension support at provisioning time:

- REST API

To disable and enable extensions when you can deploy a VM with this property:

```
"osProfile": {
    "allowExtensionOperations": false
},
```

## Next steps

Please refer to [Other base system changes since 18.04 LTS - Python 3 by default](#) for additional information.

# Migrate your IaaS resources to Azure Resource Manager by March 1, 2023

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

In 2014, we launched infrastructure as a service (IaaS) on [Azure Resource Manager](#). We've been enhancing capabilities ever since. Because Azure Resource Manager now has full IaaS capabilities and other advancements, we deprecated the management of IaaS virtual machines (VMs) through [Azure Service Manager](#) (ASM) on February 28, 2020. This functionality will be fully retired on March 1, 2023.

Today, about 90 percent of the IaaS VMs are using Azure Resource Manager. If you use IaaS resources through ASM, start planning your migration now. Complete it by March 1, 2023, to take advantage of [Azure Resource Manager](#).

VMs created using the classic deployment model will follow the [Modern Lifecycle Policy](#) for retirement.

## How does this affect me?

- As of February 28, 2020, customers who didn't utilize IaaS VMs through ASM in the month of February 2020 can no longer create VMs (classic).
- On March 1, 2023, customers will no longer be able to start IaaS VMs by using ASM. Any that are still running or allocated will be stopped and deallocated.
- On March 1, 2023, subscriptions that are not migrated to Azure Resource Manager will be informed regarding timelines for deleting any remaining VMs (classic).

This retirement does *not* affect the following Azure services and functionality:

- Storage accounts *not* used by VMs (classic)
- Virtual networks *not* used by VMs (classic)
- Other classic resources

Azure Cloud Services (classic) retirement was announced in August 2021 [here](#)

## What resources are available for this migration?

- [Microsoft Q&A](#): Microsoft and community support for migration.
- [Azure Migration Support](#): Dedicated support team for technical assistance during migration. Customers without technical support can use [free support capability](#) provided specifically for this migration.
- [Microsoft Fast Track](#): Fast track can assist eligible customers with planning & execution for this migration. [Nominate yourself](#) for DC Migration Program.
- If your company/organization has partnered with Microsoft or works with Microsoft representatives (like cloud solution architects (CSAs) or customer success account managers (CSAMs)), please work with them for additional resources for migration.

## What actions should I take?

Start planning your migration to Azure Resource Manager, today.

1. Make a list of all affected VMs:
  - The VMs of type **virtual machines (classic)** on the [Azure portal's VM pane](#) are all the affected VMs within the subscription.
  - You can also query Azure Resource Graph by using the [portal](#) or [PowerShell](#) to view the list of all flagged VMs (classic) and related information for the selected subscriptions.
  - On February 8 and September 2, 2020, we sent out emails with subject "Start planning your IaaS VM migration to Azure Resource Manager" to subscription owners. The email provides a list of all subscriptions and VMs (classic) VMs in it. Please use them to build this list.
2. [Learn more](#) about migrating your [Linux](#) and [Windows](#) VMs (classic) to Azure Resource Manager. For more information, see [Frequently asked questions about classic to Azure Resource Manager migration](#).
3. We recommend starting the planning by using the [platform support migration tool](#) to migrate your existing VMs with three easy steps: validate, prepare, and commit. The tool is designed to migrate your VMs within minimal to no downtime.
  - The first step, validate, has no impact on your existing deployment and provides a list of all unsupported scenarios for migration.
  - Go through the [list of workarounds](#) to fix your deployment and make it ready for migration.
  - Ideally after all validation errors are fixed, you should not encounter any issues during the prepare and commit steps. After the commit is successful, your deployment is live migrated to Azure Resource Manager and can then be managed through new APIs exposed by Azure Resource Manager.If the migration tool is not suitable for your migration, you can explore [other compute offerings](#) for the migration. Because there are many Azure compute offerings, and they're different from one another, we can't provide a platform-supported migration path to them.
4. For technical questions, issues, and help with adding subscriptions to the allowlist, [contact support](#).
5. Complete the migration as soon as possible to prevent business impact and to take advantage of the improved performance, security, and new features of Azure Resource Manager.

# Platform-supported migration of IaaS resources from classic to Azure Resource Manager

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

This article provides overview on platform-supported migration tool, how to migrate resources from the Azure Service Manager (ASM) also known as Classic to Resource Manager (ARM) deployment models and details how to connect resources from the two deployment models that coexist in your subscription by using virtual network site-to-site gateways. You can read more about [Azure Resource Manager features and benefits](#).

ASM supports two different compute products, Azure Virtual Machines (classic) also known as IaaS VMs & [Azure Cloud Services \(classic\)](#) also known as PaaS VMs or Web/Worker Roles. This document only talks about migrating Azure Virtual Machines (classic).

## Goal for migration

Resource Manager enables deploying complex applications through templates, configures virtual machines by using VM extensions, and incorporates access management and tagging. Azure Resource Manager includes scalable, parallel deployment for virtual machines into availability sets. The new deployment model also provides lifecycle management of compute, network, and storage independently. Finally, there's a focus on enabling security by default with the enforcement of virtual machines in a virtual network.

Almost all the features from the classic deployment model are supported for compute, network, and storage under Azure Resource Manager. To benefit from the new capabilities in Azure Resource Manager, you can migrate existing deployments from the Classic deployment model.

## Supported resources & configurations for migration

### Supported resources for migration

- Virtual Machines (Cloud Service with VMs)
- [Cloud Services \(with Web/Worker Roles\)](#)
- Availability Sets
- Storage Accounts
- Virtual Networks
- VPN Gateways
- [Express Route Gateways](#) (*in the same subscription as Virtual Network only*)
- Network Security Groups
- Route Tables
- Reserved IPs

## Supported configurations for migration

These classic IaaS resources are supported during migration

SERVICE	CONFIGURATION
Azure AD Domain Services	Virtual networks that contain Azure AD Domain services

## Supported scopes of migration

There are four different ways to complete migration of compute, network, and storage resources:

- [Migration of virtual machines \(NOT in a virtual network\)](#)
- [Migration of virtual machines \(in a virtual network\)](#)
- [Migration of storage accounts](#)
- [Migration of unattached resources](#)

### Migration of virtual machines (NOT in a virtual network)

In the Resource Manager deployment model, security is enforced for your applications by default. All VMs need to be in a virtual network in the Resource Manager model. The Azure platform restarts (`Stop`, `Deallocate`, and `Start`) the VMs as part of the migration. You have two options for the virtual networks that the Virtual Machines will be migrated to:

- You can request the platform to create a new virtual network and migrate the virtual machine into the new virtual network.
- You can migrate the virtual machine into an existing virtual network in Resource Manager.

#### NOTE

In this migration scope, both the management-plane operations and the data-plane operations may not be allowed for a period of time during the migration.

### Migration of virtual machines (in a virtual network)

For most VM configurations, only the metadata is migrating between the Classic and Resource Manager deployment models. The underlying VMs are running on the same hardware, in the same network, and with the same storage. The management-plane operations may not be allowed for a certain period of time during the migration. However, the data plane continues to work. That is, your applications running on top of VMs (classic) do not incur downtime during the migration.

The following configurations are not currently supported. If support is added in the future, some VMs in this configuration might incur downtime (go through stop, deallocate, and restart VM operations).

- You have more than one availability set in a single cloud service.
- You have one or more availability sets and VMs that are not in an availability set in a single cloud service.

#### NOTE

In this migration scope, the management plane may not be allowed for a period of time during the migration. For certain configurations as described earlier, data-plane downtime occurs.

### Migration of storage accounts

To allow seamless migration, you can deploy Resource Manager VMs in a classic storage account. With this capability, compute and network resources can and should be migrated independently of storage accounts. Once you migrate over your Virtual Machines and Virtual Network, you need to migrate over your storage accounts to complete the migration process.

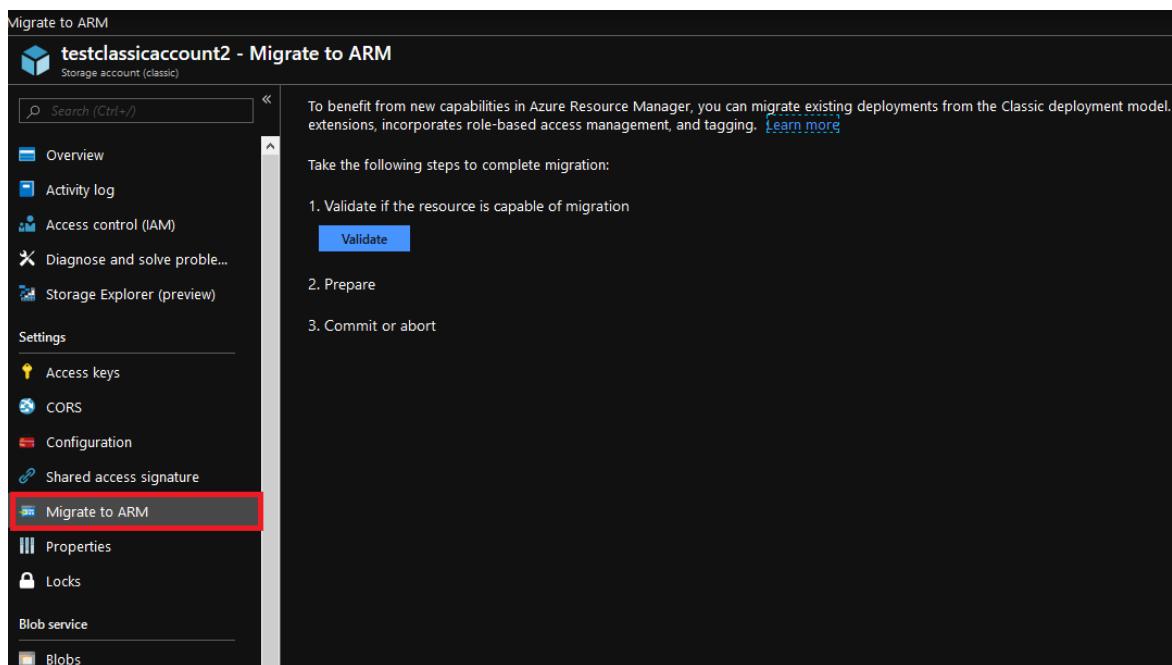
If your storage account does not have any associated disks or Virtual Machines data and only has blobs, files, tables, and queues then the migration to Azure Resource Manager can be done as a standalone migration without dependencies.

**NOTE**

The Resource Manager deployment model doesn't have the concept of Classic images and disks. When the storage account is migrated, Classic images and disks are no longer visible in the Azure portal, but the backing VHDs remain in the storage account.

The following screenshots show how to upgrade a Classic storage account to an Azure Resource Manager storage account using Azure portal:

1. Sign in to the [Azure portal](#).
2. Navigate to your storage account.
3. In the **Settings** section, click **Migrate to Azure Resource Manager**.
4. Click on **Validate** to determine migration feasibility.
5. If validation passes, click on **Prepare** to create a migrated storage account.
6. Type **yes** to confirm migration and click **Commit** to finish the migration.



The screenshot shows the 'Migrate to ARM' wizard for a storage account named 'testclassicaccount2'. The left sidebar lists options like Overview, Activity log, Access control (IAM), Diagnose and solve problems, Storage Explorer (preview), Settings (Access keys, CORS, Configuration, Shared access signature), Migrate to ARM, Properties, Locks, Blob service, and Queue service. The 'Migrate to ARM' option is selected. The main pane displays a green header bar with a checkmark and the text 'Validation passed.' Below it, a message encourages migrating to ARM for new capabilities. It lists three steps: 1. Validate if the resource is capable of migration (Validation passed, 1 storage account will be migrated), 2. Prepare (Simulate transformation, Prepare button), and 3. Commit or abort.

This screenshot continues the migration process. The validation step is complete. The preparation step shows that the storage account has been migrated to a resource group named 'testclassicaccount2-Migrated'. The commit step shows a dropdown menu set to 'yes' for confirming the migration. The 'Commit' and 'Abort' buttons are visible at the bottom.

## Migration of unattached resources

Storage Accounts with no associated disks or Virtual Machines data may be migrated independently.

Network Security Groups, Route Tables & Reserved IPs that are not attached to any Virtual Machines and Virtual Networks can also be migrated independently.

## Unsupported features and configurations

Some features and configurations are not currently supported; the following sections describe our recommendations around them.

### Unsupported features

The following features are not currently supported. You can optionally remove these settings, migrate the VMs, and then re-enable the settings in the Resource Manager deployment model.

RESOURCE PROVIDER	FEATURE	RECOMMENDATION
Compute	Unassociated virtual machine disks.	The VHD blobs behind these disks will get migrated when the Storage Account is migrated

RESOURCE PROVIDER	FEATURE	RECOMMENDATION
Compute	Virtual machine images.	The VHD blobs behind these disks will get migrated when the Storage Account is migrated
Network	Endpoint ACLs.	Remove Endpoint ACLs and retry migration.
Network	Application Gateway	Remove the Application Gateway before beginning migration and then recreate the Application Gateway once migration is complete.
Network	Virtual networks using VNet Peering.	Migrate Virtual Network to Resource Manager, then peer. Learn more about <a href="#">VNet Peering</a> .

## Unsupported configurations

The following configurations are not currently supported.

SERVICE	CONFIGURATION	RECOMMENDATION
Resource Manager	Role-Based Access Control (RBAC) for classic resources	Because the URI of the resources is modified after migration, it is recommended that you plan the RBAC policy updates that need to happen after migration.
Compute	Multiple subnets associated with a VM	Update the subnet configuration to reference only one subnet. This may require you to remove a secondary NIC (that is referring to another subnet) from the VM and reattach it after migration has completed.
Compute	Virtual machines that belong to a virtual network but don't have an explicit subnet assigned	You can optionally delete the VM.
Compute	Virtual machines that have alerts, Autoscale policies	The migration goes through and these settings are dropped. It is highly recommended that you evaluate your environment before you do the migration. Alternatively, you can reconfigure the alert settings after migration is complete.
Compute	XML VM extensions (BGInfo 1.*., Visual Studio Debugger, Web Deploy, and Remote Debugging)	This is not supported. It is recommended that you remove these extensions from the virtual machine to continue migration or they will be dropped automatically during the migration process.

Service	Configuration	Recommendation
Compute	Boot diagnostics with Premium storage	Disable Boot Diagnostics feature for the VMs before continuing with migration. You can re-enable boot diagnostics in the Resource Manager stack after the migration is complete. Additionally, blobs that are being used for screenshot and serial logs should be deleted so you are no longer charged for those blobs.
Compute	Cloud services that contain more than one availability set or multiple availability sets.	This is currently not supported. Please move the Virtual Machines to the same availability set before migrating.
Compute	VM with Microsoft Defender for Cloud extension	Microsoft Defender for Cloud automatically installs extensions on your Virtual Machines to monitor their security and raise alerts. These extensions usually get installed automatically if the Microsoft Defender for Cloud policy is enabled on the subscription. To migrate the Virtual Machines, disable the Defender for Cloud policy on the subscription, which will remove the Defender for Cloud monitoring extension from the Virtual Machines.
Compute	VM with backup or snapshot extension	These extensions are installed on a Virtual Machine configured with the Azure Backup service. While the migration of these VMs is not supported, follow the guidance in <a href="#">Frequently asked questions about classic to Azure Resource Manager migration</a> to keep backups that were taken prior to migration.
Compute	VM with Azure Site Recovery extension	These extensions are installed on a Virtual Machine configured with the Azure Site Recovery service. While the migration of storage used with Site Recovery will work, current replication will be impacted. You need to disable and enable VM replication after storage migration.
Network	Virtual networks that contain virtual machines and web/worker roles	This is currently not supported. Please move the Web/Worker roles to their own Virtual Network before migrating. Once the classic Virtual Network is migrated, the migrated Azure Resource Manager Virtual Network can be peered with the classic Virtual Network to achieve similar configuration as before.

Service	Configuration	Recommendation
Network	Classic Express Route circuits	This is currently not supported. These circuits need to be migrated to Azure Resource Manager before beginning IaaS migration. To learn more, see <a href="#">Moving ExpressRoute circuits from the classic to the Resource Manager deployment model</a> .
Azure App Service	Virtual networks that contain App Service environments	This is currently not supported.
Azure HDInsight	Virtual networks that contain HDInsight services	This is currently not supported.
Microsoft Dynamics Lifecycle Services	Virtual networks that contain virtual machines that are managed by Dynamics Lifecycle Services	This is currently not supported.
Azure API Management	Virtual networks that contain Azure API Management deployments	This is currently not supported. To migrate the IaaS VNET, change the VNET of the API Management deployment, which is a no downtime operation.

## Next steps

- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Use CLI to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

# Technical deep dive on platform-supported migration from classic to Azure Resource Manager

9/21/2022 • 13 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

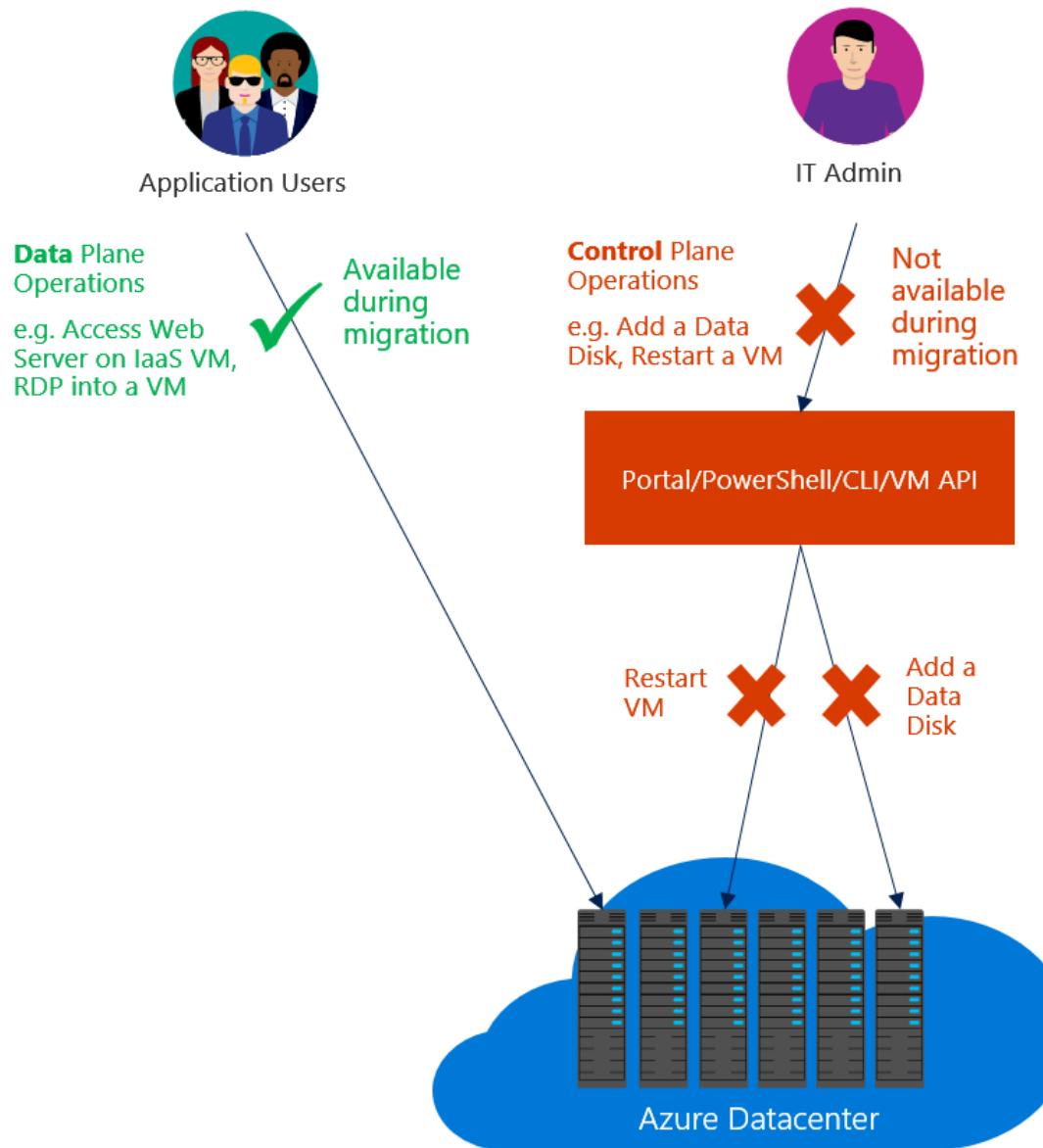
Let's take a deep-dive on migrating from the Azure classic deployment model to the Azure Resource Manager deployment model. We look at resources at a resource and feature level to help you understand how the Azure platform migrates resources between the two deployment models. For more information, please read the service announcement article: [Platform-supported migration of IaaS resources from classic to Azure Resource Manager](#).

## Migrate IaaS resources from the classic deployment model to Azure Resource Manager

First, it's important to understand the difference between data-plane and management-plane operations on the infrastructure as a service (IaaS) resources.

- *Management/control plane* describes the calls that come into the management/control plane or the API for modifying resources. For example, operations like creating a VM, restarting a VM, and updating a virtual network with a new subnet manage the running resources. They don't directly affect connecting to the VMs.
- *Data plane* (application) describes the runtime of the application itself, and involves interaction with instances that don't go through the Azure API. For example, accessing your website, or pulling data from a running SQL Server instance or a MongoDB server, are data plane or application interactions. Other examples include copying a blob from a storage account, and accessing a public IP address to use Remote Desktop Protocol (RDP) or Secure Shell (SSH) into the virtual machine. These operations keep the application running across compute, networking, and storage.

The data plane is the same between the classic deployment model and Resource Manager stacks. The difference is that during the migration process, Microsoft translates the representation of the resources from the classic deployment model to that in the Resource Manager stack. As a result, you need to use new tools, APIs, and SDKs to manage your resources in the Resource Manager stack.



#### NOTE

In some migration scenarios, the Azure platform stops, deallocates, and restarts your virtual machines. This causes a brief data-plane downtime.

## The migration experience

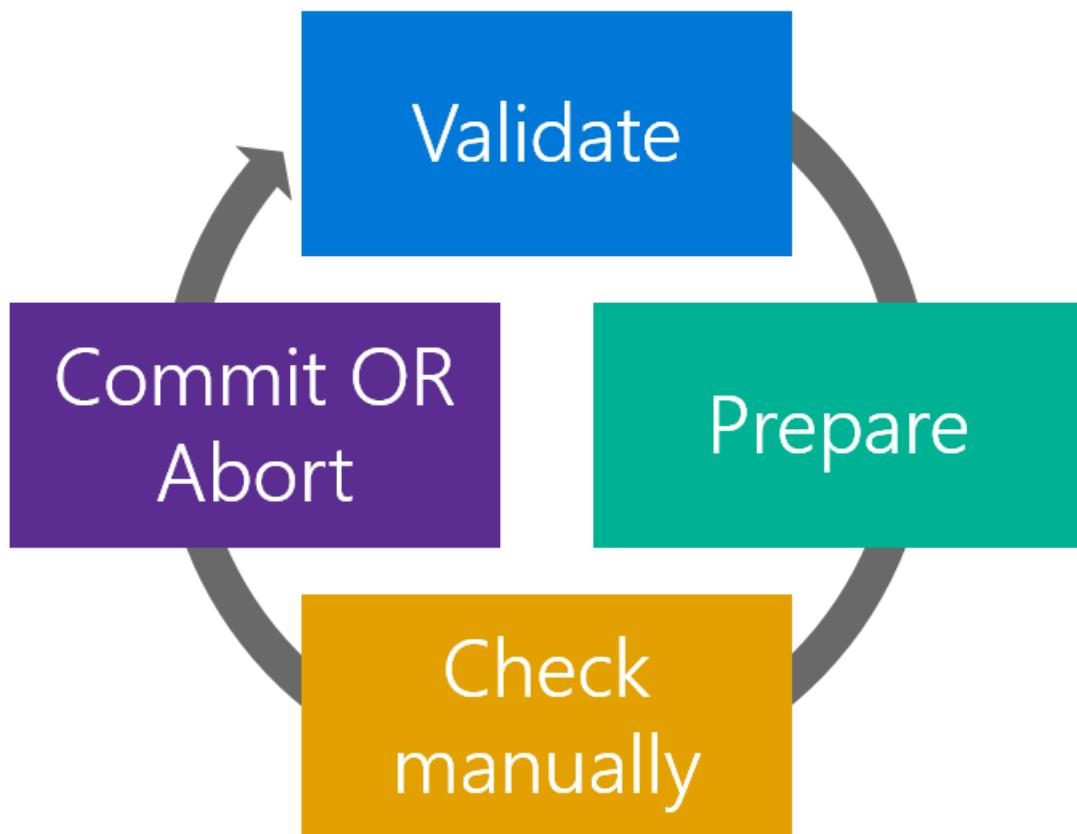
Before you start the migration:

- Ensure that the resources that you want to migrate don't use any unsupported features or configurations. Usually the platform detects these issues and generates an error.
- If you have VMs that are not in a virtual network, they are stopped and deallocated as part of the prepare operation. If you don't want to lose the public IP address, consider reserving the IP address before triggering the prepare operation. If the VMs are in a virtual network, they are not stopped and deallocated.
- Plan your migration during non-business hours to accommodate for any unexpected failures that might happen during migration.
- Download the current configuration of your VMs by using PowerShell, command-line interface (CLI) commands, or REST APIs to make it easier for validation after the prepare step is complete.
- Update your automation and operationalization scripts to handle the Resource Manager deployment model,

before you start the migration. You can optionally do GET operations when the resources are in the prepared state.

- Evaluate the Azure role-based access control (Azure RBAC) policies that are configured on the IaaS resources in the classic deployment model, and plan for after the migration is complete.

The migration workflow is as follows:



#### NOTE

The operations described in the following sections are all idempotent. If you have a problem other than an unsupported feature or a configuration error, retry the prepare, abort, or commit operation. Azure tries the action again.

## Validate

The validate operation is the first step in the migration process. The goal of this step is to analyze the state of the resources you want to migrate in the classic deployment model. The operation evaluates whether the resources are capable of migration (success or failure).

You select the virtual network or a cloud service (if it's not in a virtual network) that you want to validate for migration. If the resource is not capable of migration, Azure lists the reasons why.

#### Checks not done in the validate operation

The validate operation only analyzes the state of the resources in the classic deployment model. It can check for all failures and unsupported scenarios due to various configurations in the classic deployment model. It is not possible to check for all issues that the Azure Resource Manager stack might impose on the resources during migration. These issues are only checked when the resources undergo transformation in the next step of migration (the prepare operation). The following table lists all the issues not checked in the validate operation:

#### NETWORKING CHECKS NOT IN THE VALIDATE OPERATION

A virtual network having both ER and VPN gateways.

A virtual network gateway connection in a disconnected state.

All ER circuits are pre-migrated to Azure Resource Manager stack.

Azure Resource Manager quota checks for networking resources. For example: static public IP, dynamic public IPs, load balancer, network security groups, route tables, and network interfaces.

All load balancer rules are valid across deployment and the virtual network.

Conflicting private IPs between stop-deallocated VMs in the same virtual network.

## Prepare

The prepare operation is the second step in the migration process. The goal of this step is to simulate the transformation of the IaaS resources from the classic deployment model to Resource Manager resources. Further, the prepare operation presents this side-by-side for you to visualize.

### NOTE

Your resources in the classic deployment model are not modified during this step. It's a safe step to run if you're trying out migration.

You select the virtual network or the cloud service (if it's not a virtual network) that you want to prepare for migration.

- If the resource is not capable of migration, Azure stops the migration process and lists the reason why the prepare operation failed.
- If the resource is capable of migration, Azure locks down the management-plane operations for the resources under migration. For example, you are not able to add a data disk to a VM under migration.

Azure then starts the migration of metadata from the classic deployment model to Resource Manager for the migrating resources.

After the prepare operation is complete, you have the option of visualizing the resources in both the classic deployment model and Resource Manager. For every cloud service in the classic deployment model, the Azure platform creates a resource group name that has the pattern `cloud-service-name>-Migrated`.

### NOTE

It is not possible to select the name of a resource group created for migrated resources (that is, "-Migrated"). After migration is complete, however, you can use the move feature of Azure Resource Manager to move resources to any resource group you want. For more information, see [Move resources to new resource group or subscription](#).

The following two screenshots show the result after a successful prepare operation. The first one shows a resource group that contains the original cloud service. The second one shows the new "-Migrated" resource group that contains the equivalent Azure Resource Manager resources.

**portalmigrate**  
Resource group

Search (Ctrl+ /)

- Overview
- Activity log
- Access control (IAM)
- Tags

SETTINGS

- Quickstart
- Resource costs
- Deployments
- Properties
- Locks
- Automation script

Essentials		
Subscription name (change)	Subscription ID	
Deployments No deployments	Location East US	
<i>Filter by name...</i>		
2 items		
NAME	TYPE	LOCATION
portalmigrate	Cloud service (class...)	East US
portalmigrate	Virtual machine (cl...	East US

**portalmigrate-Migrated**  
Resource group

Search (Ctrl+ /)

- Overview
- Activity log
- Access control (IAM)
- Tags

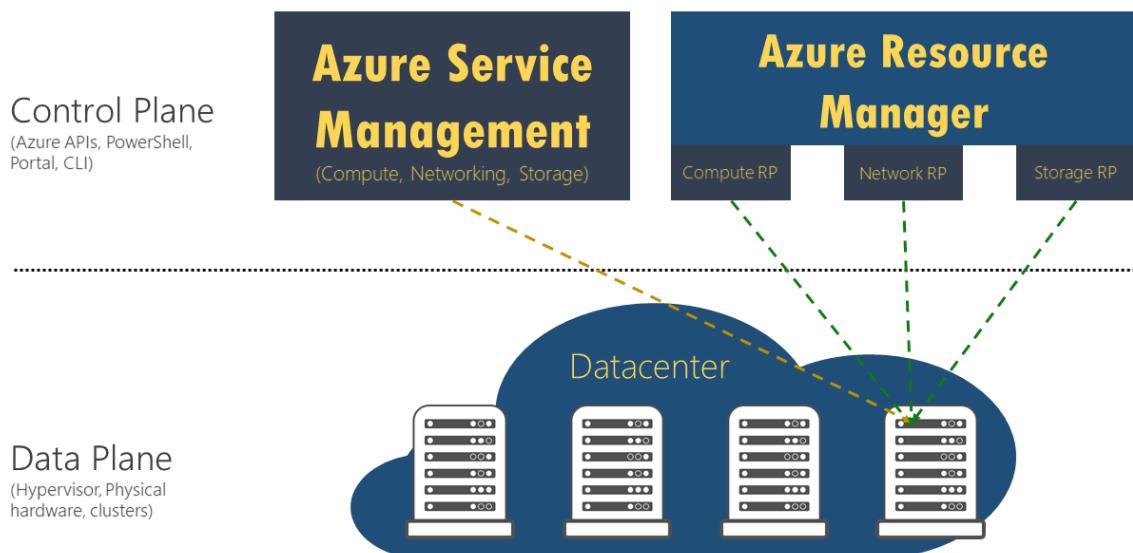
SETTINGS

- Quickstart
- Resource costs
- Deployments
- Properties
- Locks
- Automation script

Essentials		
Subscription name (change)	Subscription ID	
Deployments 2 Succeeded	Location East US	
<i>Filter by name...</i>		
5 items		
NAME	TYPE	LOCATION
portalmigrate	Virtual machine	East US
portalmigrate-PrimaryNic	Network interface	East US
portalmigrate-PrimaryVirtualIP	Public IP address	East US
portalmigrate-PublicLoadBalancer	Load balancer	East US
portalmigrate-VirtualNetwork	Virtual network	East US

Here is a behind-the-scenes look at your resources after the completion of the prepare phase. Note that the resource in the data plane is the same. It's represented in both the management plane (classic deployment model) and the control plane (Resource Manager).

# Prepare



## NOTE

VMs that are not in a virtual network in the classic deployment model are stopped and deallocated in this phase of migration.

## Check (manual or scripted)

In the check step, you have the option to use the configuration that you downloaded earlier to validate that the migration looks correct. Alternatively, you can sign in to the portal, and spot check the properties and resources to validate that metadata migration looks good.

If you are migrating a virtual network, most configuration of virtual machines is not restarted. For applications on those VMs, you can validate that the application is still running.

You can test your monitoring and operational scripts to see if the VMs are working as expected, and if your updated scripts work correctly. Only GET operations are supported when the resources are in the prepared state.

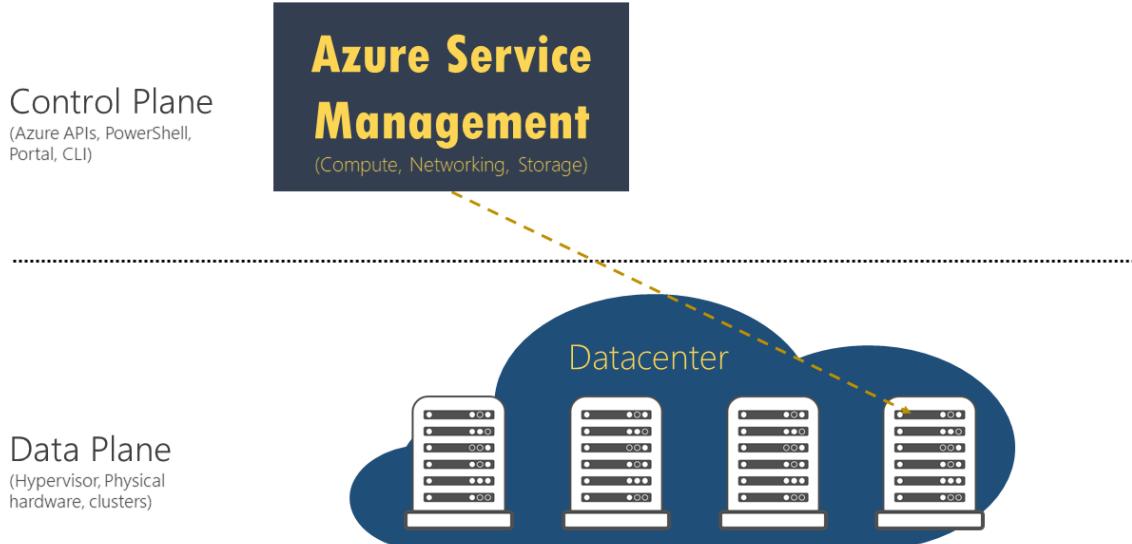
There is no set window of time before which you need to commit the migration. You can take as much time as you want in this state. However, the management plane is locked for these resources until you either abort or commit.

If you see any issues, you can always abort the migration and go back to the classic deployment model. After you go back, Azure opens the management-plane operations on the resources, so that you can resume normal operations on those VMs in the classic deployment model.

## Abort

This is an optional step if you want to revert your changes to the classic deployment model and stop the migration. This operation deletes the Resource Manager metadata (created in the prepare step) for your resources.

# Abort



## NOTE

This operation can't be done after you have triggered the commit operation.

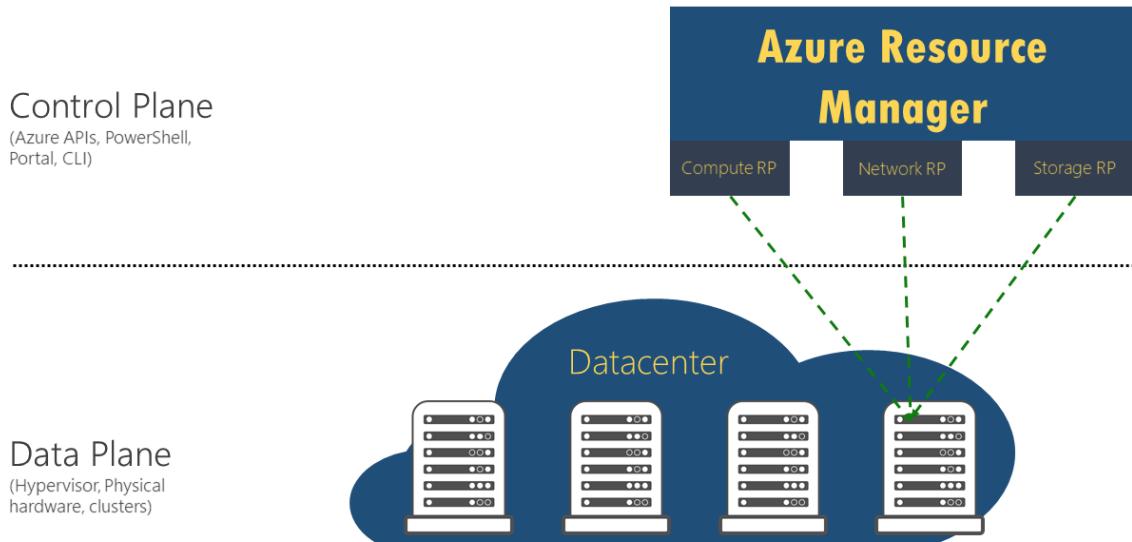
## Commit

After you finish the validation, you can commit the migration. Resources do not appear anymore in the classic deployment model, and are available only in the Resource Manager deployment model. The migrated resources can be managed only in the new portal.

## NOTE

This is an idempotent operation. If it fails, retry the operation. If it continues to fail, create a support ticket or create a forum on [Microsoft Q&A](#)

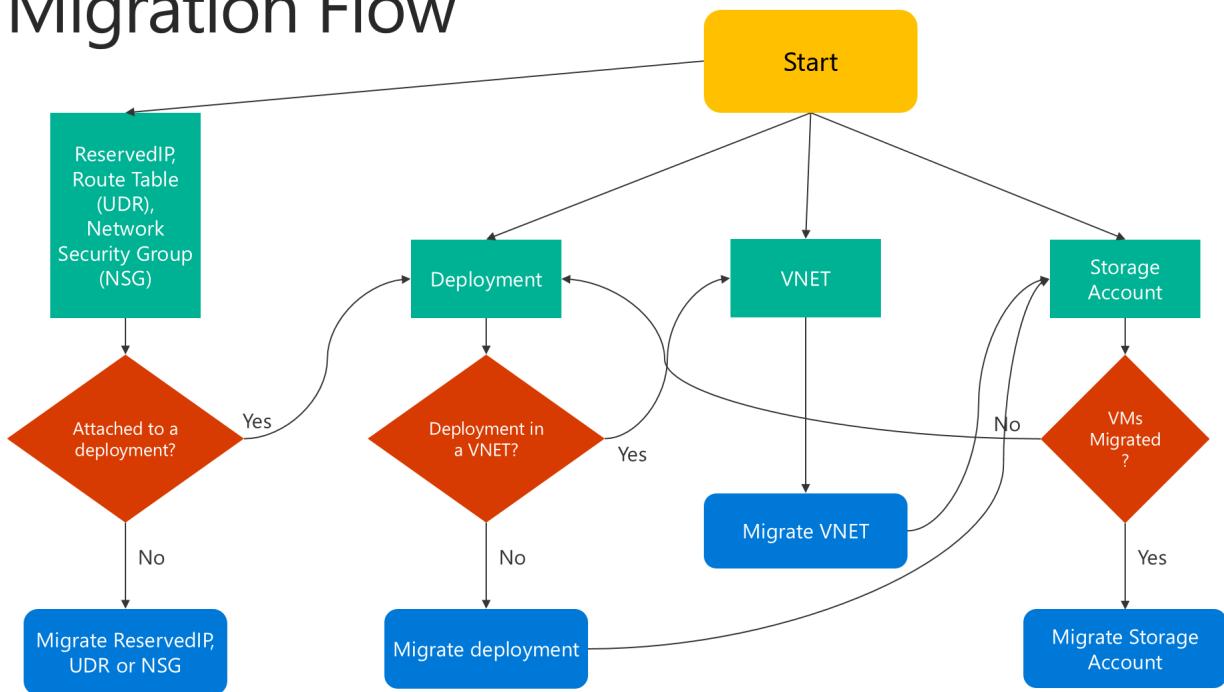
# Commit



## Migration flowchart

Here is a flowchart that shows how to proceed with migration:

# Migration Flow



## Translation of the classic deployment model to Resource Manager resources

You can find the classic deployment model and Resource Manager representations of the resources in the following table. Other features and resources are not currently supported.

CLASSIC REPRESENTATION	RESOURCE MANAGER REPRESENTATION	NOTES
Cloud service name (Hosted Service Name)	DNS name	During migration, a new resource group is created for every cloud service with the naming pattern <code>&lt;cloudservicename&gt;-migrated</code> . This resource group contains all your resources. The cloud service name becomes a DNS name that is associated with the public IP address.
Virtual machine	Virtual machine	VM-specific properties are migrated unchanged. Certain osProfile information, like computer name, is not stored in the classic deployment model, and remains empty after migration.

CLASSIC REPRESENTATION	RESOURCE MANAGER REPRESENTATION	NOTES
Disk resources attached to VM	Implicit disks attached to VM	Disks are not modeled as top-level resources in the Resource Manager deployment model. They are migrated as implicit disks under the VM. Only disks that are attached to a VM are currently supported. Resource Manager VMs can now use storage accounts in the classic deployment model, which allows the disks to be easily migrated without any updates.
VM extensions	VM extensions	All the resource extensions, except XML extensions, are migrated from the classic deployment model.
Virtual machine certificates	Certificates in Azure Key Vault	If a cloud service contains service certificates, the migration creates a new Azure key vault per cloud service, and moves the certificates into the key vault. The VMs are updated to reference the certificates from the key vault.  Do not delete the key vault. This can cause the VM to go into a failed state.
WinRM configuration	WinRM configuration under osProfile	Windows Remote Management configuration is moved unchanged, as part of the migration.
Availability-set property	Availability-set resource	Availability-set specification is a property on the VM in the classic deployment model. Availability sets become a top-level resource as part of the migration. The following configurations are not supported: multiple availability sets per cloud service, or one or more availability sets along with VMs that are not in any availability set in a cloud service.
Network configuration on a VM	Primary network interface	Network configuration on a VM is represented as the primary network interface resource after migration. For VMs that are not in a virtual network, the internal IP address changes during migration.
Multiple network interfaces on a VM	Network interfaces	If a VM has multiple network interfaces associated with it, each network interface becomes a top-level resource as part of the migration, along with all the properties.

CLASSIC REPRESENTATION	RESOURCE MANAGER REPRESENTATION	NOTES
Load-balanced endpoint set	Load balancer	In the classic deployment model, the platform assigned an implicit load balancer for every cloud service. During migration, a new load-balancer resource is created, and the load-balancing endpoint set becomes load-balancer rules.
Inbound NAT rules	Inbound NAT rules	Input endpoints defined on the VM are converted to inbound network address translation rules under the load balancer during the migration.
VIP address	Public IP address with DNS name	The virtual IP address becomes a public IP address, and is associated with the load balancer. A virtual IP can only be migrated if there is an input endpoint assigned to it. To retain the IP, you can <a href="#">convert it to Reserved IP</a> before migration. There will be downtime of about 60 seconds during this change.
Virtual network	Virtual network	The virtual network is migrated, with all its properties, to the Resource Manager deployment model. A new resource group is created with the name <code>-migrated</code> .
Reserved IPs	Public IP address with static allocation method	Reserved IPs associated with the load balancer are migrated, along with the migration of the cloud service or the virtual machine. Unassociated reserved IPs can be migrated using <a href="#">Move-AzureReservedIP</a> .
Public IP address per VM	Public IP address with dynamic allocation method	The public IP address associated with the VM is converted as a public IP address resource, with the allocation method set to dynamic.
NSGs	NSGs	Network security groups associated with a virtual machine or subnet are cloned as part of the migration to the Resource Manager deployment model. The NSG in the classic deployment model is not removed during the migration. However, the management-plane operations for the NSG are blocked when the migration is in progress. Unassociated NSGs can be migrated using <a href="#">Move-AzureNetworkSecurityGroup</a> .

CLASSIC REPRESENTATION	RESOURCE MANAGER REPRESENTATION	NOTES
DNS servers	DNS servers	DNS servers associated with a virtual network or the VM are migrated as part of the corresponding resource migration, along with all the properties.
UDRs	UDRs	User-defined routes associated with a subnet are cloned as part of the migration to the Resource Manager deployment model. The UDR in the classic deployment model is not removed during the migration. The management-plane operations for the UDR are blocked when the migration is in progress. Unassociated UDRs can be migrated using <a href="#">Move-AzureRouteTable</a> .
IP forwarding property on a VM's network configuration	IP forwarding property on the NIC	The IP forwarding property on a VM is converted to a property on the network interface during the migration.
Load balancer with multiple IPs	Load balancer with multiple public IP resources	Every public IP associated with the load balancer is converted to a public IP resource, and associated with the load balancer after migration.
Internal DNS names on the VM	Internal DNS names on the NIC	During migration, the internal DNS suffixes for the VMs are migrated to a read-only property named "InternalDomainNameSuffix" on the NIC. The suffix remains unchanged after migration, and VM resolution should continue to work as previously.
Virtual network gateway	Virtual network gateway	Virtual network gateway properties are migrated unchanged. The VIP associated with the gateway does not change either.
Local network site	Local network gateway	Local network site properties are migrated unchanged to a new resource called a local network gateway. This represents on-premises address prefixes and the remote gateway IP.

CLASSIC REPRESENTATION	RESOURCE MANAGER REPRESENTATION	NOTES
Connections references	Connection	Connectivity references between the gateway and the local network site in network configuration is represented by a new resource called Connection. All properties of connectivity reference in network configuration files are copied unchanged to the Connection resource. Connectivity between virtual networks in the classic deployment model is achieved by creating two IPsec tunnels to local network sites representing the virtual networks. This is transformed to the virtual-network-to-virtual-network connection type in the Resource Manager model, without requiring local network gateways.

## Changes to your automation and tooling after migration

As part of migrating your resources from the classic deployment model to the Resource Manager deployment model, you must update your existing automation or tooling to ensure that it continues to work after the migration.

## Next steps

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Use CLI to migrate IaaS resources from classic to Azure Resource Manager](#)
- [VPN Gateway classic to Resource Manager migration](#)
- [Migrate ExpressRoute circuits and associated virtual networks from the classic to the Resource Manager deployment model](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

# Planning for migration of IaaS resources from classic to Azure Resource Manager

9/21/2022 • 14 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

While Azure Resource Manager offers a lot of amazing features, it is critical to plan out your migration journey to make sure things go smoothly. Spending time on planning will ensure that you do not encounter issues while executing migration activities.

## NOTE

The following guidance was heavily contributed to by the Azure Customer Advisory team and Cloud Solution architects working with customers on migrating large environments. As such this document will continue to get updated as new patterns of success emerge, so check back from time to time to see if there are any new recommendations.

There are four general phases of the migration journey:



## Plan

### Technical considerations and tradeoffs

Depending on your technical requirements size, geographies and operational practices, you might want to consider:

1. Why is Azure Resource Manager desired for your organization? What are the business reasons for a migration?
2. What are the technical reasons for Azure Resource Manager? What (if any) additional Azure services would you like to leverage?
3. Which application (or sets of virtual machines) is included in the migration?
4. Which scenarios are supported with the migration API? Review the [unsupported features and configurations](#).
5. Will your operational teams now support applications/VMs in both Classic and Azure Resource Manager?
6. How (if at all) does Azure Resource Manager change your VM deployment, management, monitoring, and reporting processes? Do your deployment scripts need to be updated?
7. What is the communications plan to alert stakeholders (end users, application owners, and infrastructure owners)?
8. Depending on the complexity of the environment, should there be a maintenance period where the application is unavailable to end users and to application owners? If so, for how long?
9. What is the training plan to ensure stakeholders are knowledgeable and proficient in Azure Resource

Manager?

10. What is the program management or project management plan for the migration?
11. What are the timelines for the Azure Resource Manager migration and other related technology road maps? Are they optimally aligned?

### Patterns of success

Successful customers have detailed plans where the preceding questions are discussed, documented and governed. Ensure the migration plans are broadly communicated to sponsors and stakeholders. Equip yourself with knowledge about your migration options; reading through this migration document set below is highly recommended.

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Use CLI to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

### Pitfalls to avoid

- Failure to plan. The technology steps of this migration are proven and the outcome is predictable.
- Assumption that the platform supported migration API will account for all scenarios. Read the [unsupported features and configurations](#) to understand what scenarios are supported.
- Not planning potential application outage for end users. Plan enough buffer to adequately warn end users of potentially unavailable application time.

## Lab Test

Replicate your environment and do a test migration

#### NOTE

Exact replication of your existing environment is executed by using a community-contributed tool which is not officially supported by Microsoft Support. Therefore, it is an [optional](#) step but it is the best way to find out issues without touching your production environments. If using a community-contributed tool is not an option, then read about the Validate/Prepare/Abort Dry Run recommendation below.

Conducting a lab test of your exact scenario (compute, networking, and storage) is the best way to ensure a smooth migration. This will help ensure:

- A wholly separate lab or an existing non-production environment to test. We recommend a wholly separate lab that can be migrated repeatedly and can be destructively modified. Scripts to collect/hydrate metadata from the real subscriptions are listed below.
- It's a good idea to create the lab in a separate subscription. The reason is that the lab will be torn down repeatedly, and having a separate, isolated subscription will reduce the chance that something real will get accidentally deleted.

This can be accomplished by using the AsmMetadataParser tool. [Read more about this tool here](#)

### Patterns of success

The following were issues discovered in many of the larger migrations. This is not an exhaustive list and you should refer to the [unsupported features and configurations](#) for more detail. You may or may not encounter these technical issues but if you do solving these before attempting migration will ensure a smoother experience.

- **Do a Validate/Prepare/Abort Dry Run** - This is perhaps the most important step to ensure Classic to Azure Resource Manager migration success. The migration API has three main steps: Validate, Prepare and Commit. Validate will read the state of your classic environment and return a result of all issues. However, because some issues might exist in the Azure Resource Manager stack, Validate will not catch everything. The next step in migration process, Prepare will help expose those issues. Prepare will move the metadata from Classic to Azure Resource Manager, but will not commit the move, and will not remove or change anything on the Classic side. The dry run involves preparing the migration, then aborting (**not committing**) the migration prepare. The goal of validate/prepare/abort dry run is to see all of the metadata in the Azure Resource Manager stack, examine it (*programmatically or in Portal*), and verify that everything migrates correctly, and work through technical issues. It will also give you a sense of migration duration so you can plan for downtime accordingly. A validate/prepare/abort does not cause any user downtime; therefore, it is non-disruptive to application usage.
  - The items below will need to be solved before the dry run, but a dry run test will also safely flush out these preparation steps if they are missed. During enterprise migration, we've found the dry run to be a safe and invaluable way to ensure migration readiness.
  - When prepare is running, the control plane (Azure management operations) will be locked for the whole virtual network, so no changes can be made to VM metadata during validate/prepare/abort. But otherwise any application function (RD, VM usage, etc.) will be unaffected. Users of the VMs will not know that the dry run is being executed.
- **Express Route Circuits and VPN**. Currently Express Route Gateways with authorization links cannot be migrated without downtime. For the workaround, see [Migrate ExpressRoute circuits and associated virtual networks from the classic to the Resource Manager deployment model](#).
- **VM Extensions** - Virtual Machine extensions are potentially one of the biggest roadblocks to migrating running VMs. Remediation of VM Extensions could take upwards of 1-2 days, so plan accordingly. A working Azure agent is needed to report back VM Extension status of running VMs. If the status comes back as bad for a running VM, this will halt migration. The agent itself does not need to be in working order to enable migration, but if extensions exist on the VM, then both a working agent AND outbound internet connectivity (with DNS) will be needed for migration to move forward.
  - If connectivity to a DNS server is lost during migration, all VM Extensions except BGInfo v1.\* need to first be removed from every VM before migration prepare, and subsequently re-added back to the VM after Azure Resource Manager migration. **This is only for VMs that are running.** If the VMs are stopped deallocated, VM Extensions do not need to be removed. **Note:** Many extensions like Azure diagnostics and Defender for Cloud monitoring will reinstall themselves after migration, so removing them is not a problem.
  - In addition, make sure Network Security Groups are not restricting outbound internet access. This can happen with some Network Security Groups configurations. Outbound internet access (and DNS) is needed for VM Extensions to be migrated to Azure Resource Manager.
  - There are two versions of the BGInfo extension: v1 and v2. If the VM was created using the Azure portal or PowerShell, the VM will likely have the v1 extension on it. This extension does not need to be removed and will be skipped (not migrated) by the migration API. However, if the Classic VM was created with the new Azure portal, it will likely have the JSON-based v2 version of BGInfo, which can be migrated to Azure Resource Manager provided the agent is working and has outbound internet access (and DNS).
  - **Remediation Option 1.** If you know your VMs will not have outbound internet access, a working

DNS service, and working Azure agents on the VMs, then uninstall all VM extensions as part of the migration before Prepare, then reinstall the VM Extensions after migration.

- **Remediation Option 2.** If VM extensions are too big of a hurdle, another option is to shutdown/deallocate all VMs before migration. Migrate the deallocated VMs, then restart them on the Azure Resource Manager side. The benefit here is that VM extensions will migrate. The downside is that all public facing Virtual IPs will be lost (this may be a non-starter), and obviously the VMs will shut down causing a much greater impact on working applications.

**NOTE**

If a Microsoft Defender for Cloud policy is configured against the running VMs being migrated, the security policy needs to be stopped before removing extensions, otherwise the security monitoring extension will be reinstalled automatically on the VM after removing it.

- **Availability Sets** - For a virtual network (vNet) to be migrated to Azure Resource Manager, the Classic deployment (i.e. cloud service) contained VMs must all be in one availability set, or the VMs must all not be in any availability set. Having more than one availability set in the cloud service is not compatible with Azure Resource Manager and will halt migration. Additionally, there cannot be some VMs in an availability set, and some VMs not in an availability set. To resolve this, you will need to remediate or reshuffle your cloud service. Plan accordingly as this might be time consuming.
- **Web/Worker Role Deployments** - Cloud Services containing web and worker roles cannot migrate to Azure Resource Manager. The web/worker roles must first be removed from the virtual network before migration can start. A typical solution is to just move web/worker role instances to a separate Classic virtual network that is also linked to an ExpressRoute circuit, or to migrate the code to newer PaaS App Services (this discussion is beyond the scope of this document). In the former redeploy case, create a new Classic virtual network, move/redeploy the web/worker roles to that new virtual network, then delete the deployments from the virtual network being moved. No code changes required. The new [Virtual Network Peering](#) capability can be used to peer together the classic virtual network containing the web/worker roles and other virtual networks in the same Azure region such as the virtual network being migrated (**after virtual network migration is completed as peered virtual networks cannot be migrated**), hence providing the same capabilities with no performance loss and no latency/bandwidth penalties. Given the addition of [Virtual Network Peering](#), web/worker role deployments can now easily be mitigated and not block the migration to Azure Resource Manager.
- **Azure Resource Manager Quotas** - Azure regions have separate quotas/limits for both Classic and Azure Resource Manager. Even though in a migration scenario new hardware isn't being consumed (*we're swapping existing VMs from Classic to Azure Resource Manager*), Azure Resource Manager quotas still need to be in place with enough capacity before migration can start. Listed below are the major limits we've seen cause problems. Open a quota support ticket to raise the limits.

**NOTE**

These limits need to be raised in the same region as your current environment to be migrated.

- Network Interfaces
- Load Balancers
- Public IPs
- Static Public IPs
- Cores

- Network Security Groups
- Route Tables

You can check your current Azure Resource Manager quotas using the following commands with the latest version of Azure CLI.

#### **Compute (Cores, Availability Sets)**

```
az vm list-usage -l <azure-region> -o jsonc
```

#### **Network (Virtual Networks, Static Public IPs, Public IPs, Network Security Groups, Network Interfaces, Load Balancers, Route Tables)**

```
az network list-usages -l <azure-region> -o jsonc
```

#### **Storage (Storage Account)**

```
az storage account show-usage
```

- **Azure Resource Manager API throttling limits** - If you have a large enough environment (eg. > 400 VMs in a VNET), you might hit the default API throttling limits for writes (currently **1200 writes/hour**) in Azure Resource Manager. Before starting migration, you should raise a support ticket to increase this limit for your subscription.
- **Provisioning Timed Out VM Status** - If any VM has the status of **provisioning timed out**, this needs to be resolved pre-migration. The only way to do this is with downtime by deprovisioning/reprovisioning the VM (delete it, keep the disk, and recreate the VM).
- **RoleStateUnknown VM Status** - If migration halts due to a **role state unknown** error message, inspect the VM using the portal and ensure it is running. This error will typically go away on its own (no remediation required) after a few minutes and is often a transient type often seen during a Virtual Machine **start, stop, restart** operations. **Recommended practice:** re-try migration again after a few minutes.
- **Fabric Cluster does not exist** - In some cases, certain VMs cannot be migrated for various odd reasons. One of these known cases is if the VM was recently created (within the last week or so) and happened to land an Azure cluster that is not yet equipped for Azure Resource Manager workloads. You will get an error that says **fabric cluster does not exist** and the VM cannot be migrated. Waiting a couple of days will usually resolve this particular problem as the cluster will soon get Azure Resource Manager enabled. However, one immediate workaround is to **stop-deallocate** the VM, then continue forward with migration, and start the VM back up in Azure Resource Manager after migrating.

#### **Pitfalls to avoid**

- Do not take shortcuts and omit the validate/prepare/abort dry run migrations.
- Most, if not all, of your potential issues will surface during the validate/prepare/abort steps.

## **Migration**

#### **Technical considerations and tradeoffs**

Now you are ready because you have worked through the known issues with your environment.

For the real migrations, you might want to consider:

1. Plan and schedule the virtual network (smallest unit of migration) with increasing priority. Do the simple virtual networks first, and progress with the more complicated virtual networks.
2. Most customers will have non-production and production environments. Schedule production last.
3. (OPTIONAL) Schedule a maintenance downtime with plenty of buffer in case unexpected issues arise.
4. Communicate with and align with your support teams in case issues arise.

### Patterns of success

The technical guidance from the Lab Test section above should be considered and mitigated prior to a real migration. With adequate testing, the migration is actually a non-event. For production environments, it might be helpful to have additional support, such as a trusted Microsoft partner or Microsoft Premier services.

### Pitfalls to avoid

Not fully testing may cause issues and delay in the migration.

## Beyond Migration

### Technical considerations and tradeoffs

Now that you are in Azure Resource Manager, maximize the platform. Read the [overview of Azure Resource Manager](#) to find out about additional benefits.

Things to consider:

- Bundling the migration with other activities. Most customers opt for an application maintenance window. If so, you might want to use this downtime to enable other Azure Resource Manager capabilities like encryption and migration to Managed Disks.
- Revisit the technical and business reasons for Azure Resource Manager; enable the additional services available only on Azure Resource Manager that apply to your environment.
- Modernize your environment with PaaS services.

### Patterns of success

Be purposeful on what services you now want to enable in Azure Resource Manager. Many customers find the below compelling for their Azure environments:

- [Azure role-based access control \(Azure RBAC\)](#).
- [Azure Resource Manager templates for easier and more controlled deployment](#).
- [Tags](#).
- [Activity Control](#)
- [Azure Policies](#)

### Pitfalls to avoid

Remember why you started this Classic to Azure Resource Manager migration journey. What were the original business reasons? Did you achieve the business reason?

## Next steps

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

Manager

# Migrate IaaS resources from classic to Azure Resource Manager by using Azure CLI

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

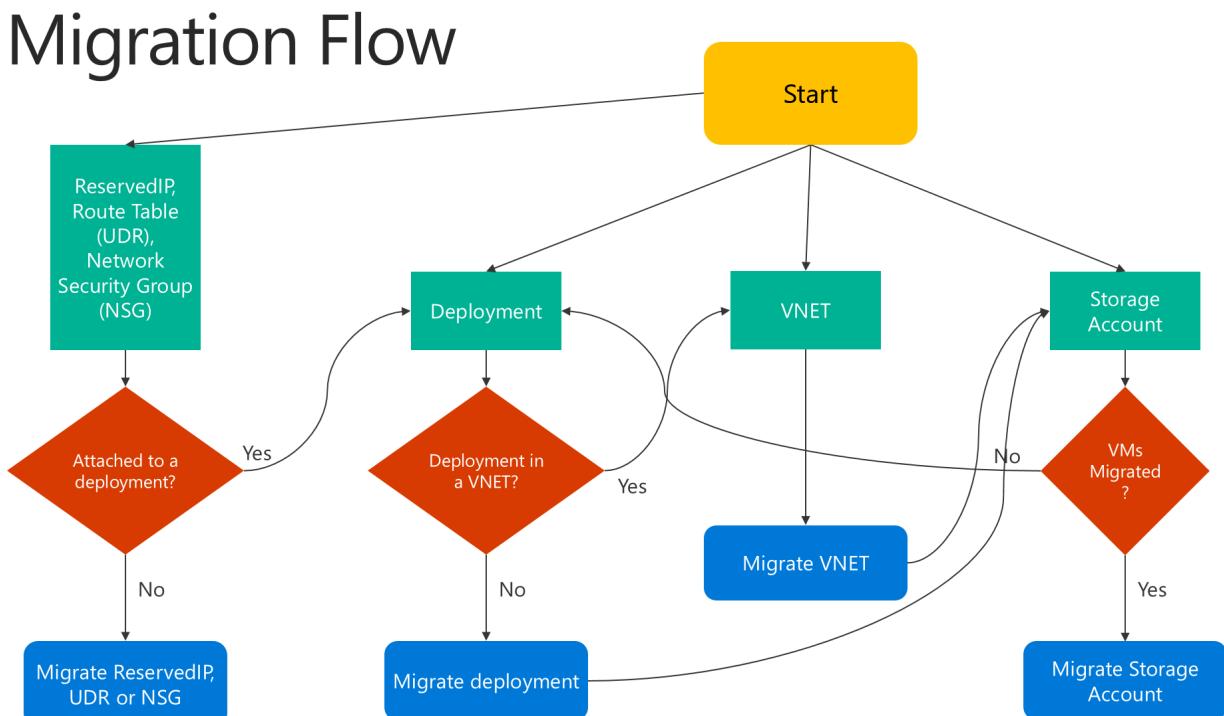
Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

These steps show you how to use CLI commands to migrate infrastructure as a service (IaaS) resources from the classic deployment model to the Azure Resource Manager deployment model. The article requires the [Azure classic CLI](#). Since Azure CLI only applies to Azure Resource Manager resources, it cannot be used for this migration.

## NOTE

All the operations described here are idempotent. If you have a problem other than an unsupported feature or a configuration error, we recommend that you retry the prepare, abort, or commit operation. The platform will then try the action again.

Here is a flowchart to identify the order in which steps need to be executed during a migration process



## Step 1: Prepare for migration

Here are a few best practices that we recommend as you evaluate migrating IaaS resources from classic to Resource Manager:

- Read through the [list of unsupported configurations or features](#). If you have virtual machines that use unsupported configurations or features, we recommend that you wait for the feature/configuration support to be announced. Alternatively, you can remove that feature or move out of that configuration to enable migration if it suits your needs.
- If you have automated scripts that deploy your infrastructure and applications today, try to create a similar test setup by using those scripts for migration. Alternatively, you can set up sample environments by using the Azure portal.

#### IMPORTANT

Application Gateways are not currently supported for migration from classic to Resource Manager. To migrate a classic virtual network with an Application gateway, remove the gateway before running a Prepare operation to move the network. After you complete the migration, reconnect the gateway in Azure Resource Manager.

ExpressRoute gateways connecting to ExpressRoute circuits in another subscription cannot be migrated automatically. In such cases, remove the ExpressRoute gateway, migrate the virtual network and recreate the gateway. Please see [Migrate ExpressRoute circuits and associated virtual networks from the classic to the Resource Manager deployment model](#) for more information.

## Step 2: Set your subscription and register the provider

For migration scenarios, you need to set up your environment for both classic and Resource Manager. [Install the Azure classic CLI](#) and [select your subscription](#).

Sign-in to your account.

```
azure login
```

Select the Azure subscription by using the following command.

```
azure account set "<azure-subscription-name>"
```

#### NOTE

Registration is a one time step but it needs to be done once before attempting migration. Without registering you'll see the following error message

*BadRequest : Subscription is not registered for migration.*

Register with the migration resource provider by using the following command. Note that in some cases, this command times out. However, the registration will be successful.

```
azure provider register Microsoft.ClassicInfrastructureMigrate
```

Please wait five minutes for the registration to finish. You can check the status of the approval by using the following command. Make sure that RegistrationState is `Registered` before you proceed.

```
azure provider show Microsoft.ClassicInfrastructureMigrate
```

Now switch CLI to the `asm` mode.

```
azure config mode asm
```

## Step 3: Make sure you have enough Azure Resource Manager Virtual Machine vCPUs in the Azure region of your current deployment or VNET

For this step you'll need to switch to `arm` mode. Do this with the following command.

```
azure config mode arm
```

You can use the following CLI command to check the current number of vCPUs you have in Azure Resource Manager. To learn more about vCPU quotas, see [Limits and the Azure Resource Manager](#).

```
azure vm list-usage -l "<Your VNET or Deployment's Azure region"
```

Once you're done verifying this step, you can switch back to `asm` mode.

```
azure config mode asm
```

## Step 4: Option 1 - Migrate virtual machines in a cloud service

Get the list of cloud services by using the following command, and then pick the cloud service that you want to migrate. Note that if the VMs in the cloud service are in a virtual network or if they have web/worker roles, you will get an error message.

```
azure service list
```

Run the following command to get the deployment name for the cloud service from the verbose output. In most cases, the deployment name is the same as the cloud service name.

```
azure service show <serviceName> -vv
```

First, validate if you can migrate the cloud service using the following commands:

```
azure service deployment validate-migration <serviceName> <deploymentName> new "" "" ""
```

Prepare the virtual machines in the cloud service for migration. You have two options to choose from.

If you want to migrate the VMs to a platform-created virtual network, use the following command.

```
azure service deployment prepare-migration <serviceName> <deploymentName> new "" "" ""
```

If you want to migrate to an existing virtual network in the Resource Manager deployment model, use the following command.

```
azure service deployment prepare-migration <serviceName> <deploymentName> existing
<destinationVNETResourceGroupName> <subnetName> <vnetName>
```

After the prepare operation is successful, you can look through the verbose output to get the migration state of the VMs and ensure that they are in the `Prepared` state.

```
azure vm show <vmName> -vv
```

Check the configuration for the prepared resources by using either CLI or the Azure portal. If you are not ready for migration and you want to go back to the old state, use the following command.

```
azure service deployment abort-migration <serviceName> <deploymentName>
```

If the prepared configuration looks good, you can move forward and commit the resources by using the following command.

```
azure service deployment commit-migration <serviceName> <deploymentName>
```

## Step 4: Option 2 - Migrate virtual machines in a virtual network

Pick the virtual network that you want to migrate. Note that if the virtual network contains web/worker roles or VMs with unsupported configurations, you will get a validation error message.

Get all the virtual networks in the subscription by using the following command.

```
azure network vnet list
```

The output will look something like this:

```
info: Executing command network vnet list
+ Looking up the virtual network sites
data: Name                           Location  Affinity gr
data: -----
data: Group classicubuntu16 classicubuntu16   East US
data: Group Group LinuxHost           East US
data: Group LinuxRG LinuxRG          East US
data: Group SUSEClassicRG SUSEClassicRG    East US
info: network vnet list command OK
```

In the above example, the **virtualNetworkName** is the entire name "**Group classicubuntu16 classicubuntu16**".

First, validate if you can migrate the virtual network using the following command:

```
azure network vnet validate-migration <virtualNetworkName>
```

Prepare the virtual network of your choice for migration by using the following command.

```
azure network vnet prepare-migration <virtualNetworkName>
```

Check the configuration for the prepared virtual machines by using either CLI or the Azure portal. If you are not ready for migration and you want to go back to the old state, use the following command.

```
azure network vnet abort-migration <virtualNetworkName>
```

If the prepared configuration looks good, you can move forward and commit the resources by using the

following command.

```
azure network vnet commit-migration <virtualNetworkName>
```

## Step 5: Migrate a storage account

Once you're done migrating the virtual machines, we recommend you migrate the storage account.

Prepare the storage account for migration by using the following command

```
azure storage account prepare-migration <storageAccountName>
```

Check the configuration for the prepared storage account by using either CLI or the Azure portal. If you are not ready for migration and you want to go back to the old state, use the following command.

```
azure storage account abort-migration <storageAccountName>
```

If the prepared configuration looks good, you can move forward and commit the resources by using the following command.

```
azure storage account commit-migration <storageAccountName>
```

## Next steps

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

# Migrate IaaS resources from classic to Azure Resource Manager by using PowerShell

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

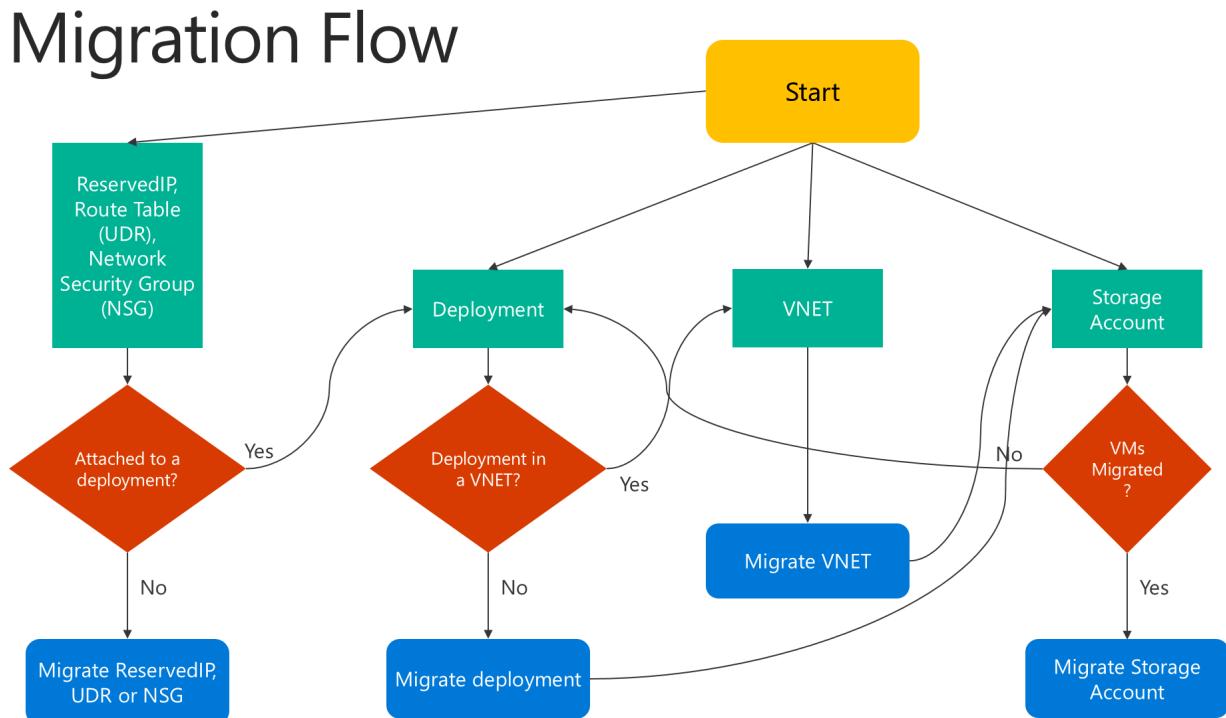
Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

These steps show you how to use Azure PowerShell commands to migrate infrastructure as a service (IaaS) resources from the classic deployment model to the Azure Resource Manager deployment model.

If you want, you can also migrate resources by using the [Azure CLI](#).

- For background on supported migration scenarios, see [Platform-supported migration of IaaS resources from classic to Azure Resource Manager](#).
- For detailed guidance and a migration walkthrough, see [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#).
- Review the [most common migration errors](#).

Here's a flowchart to identify the order in which steps need to be executed during a migration process.



## Step 1: Plan for migration

Here are a few best practices that we recommend as you evaluate whether to migrate IaaS resources from classic to Resource Manager:

- Read through the [supported and unsupported features and configurations](#). If you have virtual machines that use unsupported configurations or features, wait for the configuration or feature support to be announced. Alternatively, if it suits your needs, remove that feature or move out of that configuration to enable migration.
- If you have automated scripts that deploy your infrastructure and applications today, try to create a similar test setup by using those scripts for migration. Alternatively, you can set up sample environments by using the Azure portal.

#### IMPORTANT

Application gateways aren't currently supported for migration from classic to Resource Manager. To migrate a virtual network with an application gateway, remove the gateway before you run a Prepare operation to move the network. After you complete the migration, reconnect the gateway in Azure Resource Manager.

Azure ExpressRoute gateways that connect to ExpressRoute circuits in another subscription can't be migrated automatically. In such cases, remove the ExpressRoute gateway, migrate the virtual network, and re-create the gateway. For more information, see [Migrate ExpressRoute circuits and associated virtual networks from the classic to the Resource Manager deployment model](#).

## Step 2: Install the latest version of PowerShell

There are two main options to install Azure PowerShell: [PowerShell Gallery](#) or [Web Platform Installer \(WebPI\)](#). WebPI receives monthly updates. PowerShell Gallery receives updates on a continuous basis. This article is based on Azure PowerShell version 2.1.0.

For installation instructions, see [How to install and configure Azure PowerShell](#).

## Step 3: Ensure that you're an administrator for the subscription

To perform this migration, you must be added as a coadministrator for the subscription in the [Azure portal](#).

1. Sign in to the [Azure portal](#).
2. On the **Hub** menu, select **Subscription**. If you don't see it, select **All services**.
3. Find the appropriate subscription entry, and then look at the **MY ROLE** field. For a coadministrator, the value should be *Account admin*.

If you're not able to add a co-administrator, contact a service administrator or co-administrator for the subscription to get yourself added.

## Step 4: Set your subscription, and sign up for migration

First, start a PowerShell prompt. For migration, set up your environment for both classic and Resource Manager.

Sign in to your account for the Resource Manager model.

```
Connect-AzAccount
```

Get the available subscriptions by using the following command:

```
Get-AzSubscription | Sort Name | Select Name
```

Set your Azure subscription for the current session. This example sets the default subscription name to **My Azure Subscription**. Replace the example subscription name with your own.

```
Select-AzSubscription -SubscriptionName "My Azure Subscription"
```

#### NOTE

Registration is a one-time step, but you must do it once before you attempt migration. Without registering, you see the following error message:

*BadRequest : Subscription is not registered for migration.*

Register with the migration resource provider by using the following command:

```
Register-AzResourceProvider -ProviderNamespace Microsoft.ClassicInfrastructureMigrate
```

Wait five minutes for the registration to finish. Check the status of the approval by using the following command:

```
Get-AzResourceProvider -ProviderNamespace Microsoft.ClassicInfrastructureMigrate
```

Make sure that RegistrationState is `Registered` before you proceed.

Before switching to the classic deployment model, make sure that you have enough Azure Resource Manager virtual machine vCPUs in the Azure region of your current deployment or virtual network. You can use the following PowerShell command to check the current number of vCPUs you have in Azure Resource Manager. To learn more about vCPU quotas, see [Limits and the Azure Resource Manager](#).

This example checks the availability in the **West US** region. Replace the example region name with your own.

```
Get-AzVMUsage -Location "West US"
```

Now, sign in to your account for the classic deployment model.

```
Add-AzureAccount
```

Get the available subscriptions by using the following command:

```
Get-AzureSubscription | Sort SubscriptionName | Select SubscriptionName
```

Set your Azure subscription for the current session. This example sets the default subscription to **My Azure Subscription**. Replace the example subscription name with your own.

```
Select-AzureSubscription -SubscriptionName "My Azure Subscription"
```

## Step 5: Run commands to migrate your IaaS resources

- [Migrate VMs in a cloud service \(not in a virtual network\)](#)
- [Migrate VMs in a virtual network](#)
- [Migrate a storage account](#)

**NOTE**

All the operations described here are idempotent. If you have a problem other than an unsupported feature or a configuration error, we recommend that you retry the prepare, abort, or commit operation. The platform then tries the action again.

**Step 5.1: Option 1 - Migrate virtual machines in a cloud service (not in a virtual network)**

Get the list of cloud services by using the following command. Then pick the cloud service that you want to migrate. If the VMs in the cloud service are in a virtual network or if they have web or worker roles, the command returns an error message.

```
Get-AzureService | ft Servicename
```

Get the deployment name for the cloud service. In this example, the service name is **My Service**. Replace the example service name with your own service name.

```
$serviceName = "My Service"  
$deployment = Get-AzureDeployment -ServiceName $serviceName  
$deploymentName = $deployment.DeploymentName
```

Prepare the virtual machines in the cloud service for migration. You have two options to choose from.

- **Option 1: Migrate the VMs to a platform-created virtual network.**

First, validate that you can migrate the cloud service by using the following commands:

```
$validate = Move-AzureService -Validate -ServiceName $serviceName `  
    -DeploymentName $deploymentName -CreateNewVirtualNetwork  
$validate.ValidationMessages
```

The following command displays any warnings and errors that block migration. If validation messages do not contain message of type error, you can move on to the Prepare step.

```
Move-AzureService -Prepare -ServiceName $serviceName `  
    -DeploymentName $deploymentName -CreateNewVirtualNetwork
```

- **Option 2: Migrate to an existing virtual network in the Resource Manager deployment model.**

This example sets the resource group name to **myResourceGroup**, the virtual network name to **myVirtualNetwork**, and the subnet name to **mySubNet**. Replace the names in the example with the names of your own resources.

```
$existingVnetRGName = "myResourceGroup"  
$vnetName = "myVirtualNetwork"  
$subnetName = "mySubNet"
```

First, validate that you can migrate the virtual network by using the following command:

```
$validate = Move-AzureService -Validate -ServiceName $serviceName `  
    -DeploymentName $deploymentName -UseExistingVirtualNetwork -VirtualNetworkResourceGroupName  
$existingVnetRGName -VirtualNetworkName $vnetName -SubnetName $subnetName  
$validate.ValidationMessages
```

The following command displays any warnings and errors that block migration. If validation messages do not contain errors, you can proceed with the following Prepare step:

```
Move-AzureService -Prepare -ServiceName $serviceName -DeploymentName $deploymentName `  
    -UseExistingVirtualNetwork -VirtualNetworkResourceGroupName $existingVnetRGName `  
    -VirtualNetworkName $vnetName -SubnetName $subnetName
```

After the Prepare operation succeeds with either of the preceding options, query the migration state of the VMs. Ensure that they're in the **Prepared** state.

This example sets the VM name to **myVM**. Replace the example name with your own VM name.

```
$vmName = "myVM"  
$vm = Get-AzureVM -ServiceName $serviceName -Name $vmName  
$vm.VM.MigrationState
```

Check the configuration for the prepared resources by using either PowerShell or the Azure portal. If you're not ready for migration and you want to go back to the old state, use the following command:

```
Move-AzureService -Abort -ServiceName $serviceName -DeploymentName $deploymentName
```

If the prepared configuration looks good, you can move forward and commit the resources by using the following command:

```
Move-AzureService -Commit -ServiceName $serviceName -DeploymentName $deploymentName
```

### Step 5.1: Option 2 - Migrate virtual machines in a virtual network

To migrate virtual machines in a virtual network, you migrate the virtual network. The virtual machines automatically migrate with the virtual network. Pick the virtual network that you want to migrate.

#### NOTE

Migrate a single virtual machine created using the classic deployment model by creating a new Resource Manager virtual machine with Managed Disks by using the VHD (OS and data) files of the virtual machine.

#### NOTE

The virtual network name might be different from what is shown in the new portal. The new Azure portal displays the name as `[vnet-name]`, but the actual virtual network name is of type `Group [resource-group-name] [vnet-name]`. Before you start the migration, look up the actual virtual network name by using the command

```
Get-AzureVnetSite | Select -Property Name
```

This example sets the virtual network name to **myVnet**. Replace the example virtual network name with your own.

```
$vnetName = "myVnet"
```

#### NOTE

If the virtual network contains web or worker roles, or VMs with unsupported configurations, you get a validation error message.

First, validate that you can migrate the virtual network by using the following command:

```
Move-AzureVirtualNetwork -Validate -VirtualNetworkName $vnetName
```

The following command displays any warnings and errors that block migration. If validation is successful, you can proceed with the following Prepare step:

```
Move-AzureVirtualNetwork -Prepare -VirtualNetworkName $vnetName
```

Check the configuration for the prepared virtual machines by using either Azure PowerShell or the Azure portal. If you're not ready for migration and you want to go back to the old state, use the following command:

```
Move-AzureVirtualNetwork -Abort -VirtualNetworkName $vnetName
```

If the prepared configuration looks good, you can move forward and commit the resources by using the following command:

```
Move-AzureVirtualNetwork -Commit -VirtualNetworkName $vnetName
```

### Step 5.2: Migrate a storage account

After you're done migrating the virtual machines, perform the following prerequisite checks before you migrate the storage accounts.

#### NOTE

If your storage account has no associated disks or VM data, you can skip directly to the "Validate storage accounts and start migration" section. Also note that deleting the classic disks, VM images or OS images does not remove the source VHD files in the storage account. However, it does break the lease on those VHD files so that they can be reused to create ARM disks or images after migration.

- Prerequisite checks if you migrated any VMs or your storage account has disk resources:

- Migrate virtual machines whose disks are stored in the storage account.

The following command returns RoleName and DiskName properties of all the VM disks in the storage account. RoleName is the name of the virtual machine to which a disk is attached. If this command returns disks, then ensure that virtual machines to which these disks are attached are migrated before you migrate the storage account.

```
$storageAccountName = 'yourStorageAccountName'  
Get-AzureDisk | where-Object {$_.MediaLink.Host.Contains($storageAccountName)} | Select-  
Object -ExpandProperty AttachedTo -Property `  
DiskName | Format-List -Property RoleName, DiskName
```

- Delete unattached VM disks stored in the storage account.

Find unattached VM disks in the storage account by using the following command:

```
$storageAccountName = 'yourStorageAccountName'  
Get-AzureDisk | where-Object {$_.MediaLink.Host.Contains($storageAccountName)} | Where-  
Object -Property AttachedTo -EQ $null | Format-List -Property DiskName
```

If the previous command returns disks, delete these disks by using the following command:

```
Remove-AzureDisk -DiskName 'yourDiskName'
```

- Delete VM images stored in the storage account.

The following command returns all the VM images with OS disks stored in the storage account.

```
Get-AzureVmImage | Where-Object { $_.OSDiskConfiguration.MediaLink -ne $null -and  
$_.OSDiskConfiguration.MediaLink.Host.Contains($storageAccountName)`  
} | Select-Object -Property ImageName, ImageLabel
```

The following command returns all the VM images with data disks stored in the storage account.

```
Get-AzureVmImage | Where-Object {$_.DataDiskConfigurations -ne $null `  
-and ($_.DataDiskConfigurations | Where-Object  
{$_.MediaLink -ne $null -and $_.MediaLink.Host.Contains($storageAccountName)}).Count -gt 0 `  
} | Select-Object -Property ImageName, ImageLabel
```

Delete all the VM images returned by the previous commands by using this command:

```
Remove-AzureVMImage -ImageName 'yourImageName'
```

- Validate storage accounts and start migration.

Validate each storage account for migration by using the following command. In this example, the storage account name is **myStorageAccount**. Replace the example name with the name of your own storage account.

```
$storageAccountName = "myStorageAccount"  
Move-AzureStorageAccount -Validate -StorageAccountName $storageAccountName
```

The next step is to prepare the storage account for migration.

```
$storageAccountName = "myStorageAccount"  
Move-AzureStorageAccount -Prepare -StorageAccountName $storageAccountName
```

Check the configuration for the prepared storage account by using either Azure PowerShell or the Azure portal. If you're not ready for migration and you want to go back to the old state, use the following command:

```
Move-AzureStorageAccount -Abort -StorageAccountName $storageAccountName
```

If the prepared configuration looks good, you can move forward and commit the resources by using the following command:

```
Move-AzureStorageAccount -Commit -StorageAccountName $storageAccountName
```

## Next steps

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use CLI to migrate IaaS resources from classic to Azure Resource Manager](#)
- [Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager](#)
- [Review most common migration errors](#)
- [Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager](#)

# Errors that commonly occur during Classic to Azure Resource Manager migration

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

This article catalogs the most common errors and mitigations during the migration of IaaS resources from Azure classic deployment model to the Azure Resource Manager stack.

## List of errors

ERROR STRING	MITIGATION
Internal server error	<p>In some cases, this is a transient error that goes away with a retry. If it continues to persist, <a href="#">contact Azure support</a> as it needs investigation of platform logs.</p> <p><b>NOTE:</b> Once the incident is tracked by the support team, please don't attempt any self-mitigation as this might have unintended consequences on your environment.</p>
Migration isn't supported for Deployment {deployment-name} in HostedService {hosted-service-name} because it's a PaaS deployment (Web/Worker).	<p>This happens when a deployment contains a web/worker role. Since migration is only supported for Virtual Machines, please remove the web/worker role from the deployment and try migration again.</p>
Template {template-name} deployment failed. CorrelationId={guid}	<p>In the backend of migration service, we use Azure Resource Manager templates to create resources in the Azure Resource Manager stack. Since templates are idempotent, usually you can safely retry the migration operation to get past this error. If this error continues to persist, please <a href="#">contact Azure support</a> and give them the CorrelationId.</p> <p><b>NOTE:</b> Once the incident is tracked by the support team, please don't attempt any self-mitigation as this might have unintended consequences on your environment.</p>
The virtual network {virtual-network-name} doesn't exist.	<p>This can happen if you created the Virtual Network in the new Azure portal. The actual Virtual Network name follows the pattern "Group * &lt;VNET name&gt;"</p>
VM {vm-name} in HostedService {hosted-service-name} contains Extension {extension-name} which isn't supported in Azure Resource Manager. It's recommended to uninstall it from the VM before continuing with migration.	<p>XML extensions such as BGInfo 1.* aren't supported in Azure Resource Manager. Therefore, these extensions can't be migrated. If these extensions are left installed on the virtual machine, they're automatically uninstalled before completing the migration.</p>

ERROR STRING	MITIGATION
<p>VM {vm-name} in HostedService {hosted-service-name} contains Extension VMSnapshot/VMSnapshotLinux, which is currently not supported for Migration. Uninstall it from the VM and add it back using Azure Resource Manager after the Migration is Complete</p>	<p>This is the scenario where the virtual machine is configured for Azure Backup. Since this is currently an unsupported scenario, please follow the workaround at <a href="https://aka.ms/vmbackupmigration">https://aka.ms/vmbackupmigration</a></p>
<p>VM {vm-name} in HostedService {hosted-service-name} contains Extension {extension-name} whose Status isn't being reported from the VM. Hence, this VM can't be migrated. Ensure that the Extension status is being reported or uninstall the extension from the VM and retry migration.</p>	<p>Azure guest agent &amp; VM Extensions need outbound internet access to the VM storage account to populate their status. Common causes of status failure include</p> <ul style="list-style-type: none"> <li>• a Network Security Group that blocks outbound access to the internet</li> <li>• If the VNET has on premises DNS servers and DNS connectivity is lost</li> </ul>
<p>VM {vm-name} in HostedService {hosted-service-name} contains Extension {extension-name} reporting Handler Status: {handler-status}. Hence, the VM can't be migrated. Ensure that the Extension handler status being reported is {handler-status} or uninstall it from the VM and retry migration.</p>	<p>If you continue to see an unsupported status, you can uninstall the extensions to skip this check and move forward with migration.</p>
<p>VM Agent for VM {vm-name} in HostedService {hosted-service-name} is reporting the overall agent status as Not Ready. Hence, the VM may not be migrated, if it has a migratable extension. Ensure that the VM Agent is reporting overall agent status as Ready. Refer to <a href="https://aka.ms/classiciaasmigrationfaqs">https://aka.ms/classiciaasmigrationfaqs</a>.</p>	
<p>Migration isn't supported for Deployment {deployment-name} in HostedService {hosted-service-name} because it has multiple Availability Sets.</p>	<p>Currently, only hosted services that have 1 or less Availability sets can be migrated. To work around this problem, move the additional availability sets, and Virtual machines in those availability sets, to a different hosted service.</p>
<p>Migration isn't supported for Deployment {deployment-name} in HostedService {hosted-service-name} because it has VMs that are not part of the Availability Set even though the HostedService contains one.</p>	<p>The workaround for this scenario is to either move all the virtual machines into a single Availability set or remove all Virtual machines from the Availability set in the hosted service.</p>
<p>Storage account/HostedService/Virtual Network {virtual-network-name} is in the process of being migrated and hence cannot be changed</p>	<p>This error happens when the "Prepare" migration operation has been completed on the resource and an operation that would make a change to the resource is triggered. Because of the lock on the management plane after "Prepare" operation, any changes to the resource are blocked. To unlock the management plane, you can run the "Commit" migration operation to complete migration or the "Abort" migration operation to roll back the "Prepare" operation.</p>
<p>Migration isn't allowed for HostedService {hosted-service-name} because it has VM {vm-name} in State: RoleStateUnknown. Migration is allowed only when the VM is in one of the following states - Running, Stopped, Stopped Deallocated.</p>	<p>The VM might be undergoing through a state transition, which usually happens when during an update operation on the HostedService such as a reboot, extension installation etc. It is recommended for the update operation to complete on the HostedService before trying migration.</p>
<p>Deployment {deployment-name} in HostedService {hosted-service-name} contains a VM {vm-name} with Data Disk {data-disk-name} whose physical blob size {size-of-the-vhd-blob-backing-the-data-disk} bytes doesn't match the VM Data Disk logical size {size-of-the-data-disk-specified-in-the-vm-api} bytes. Migration will proceed without specifying a size for the data disk for the Azure Resource Manager VM.</p>	<p>This error happens if you've resized the VHD blob without updating the size in the VM API model. Detailed mitigation steps are outlined <a href="#">below</a>.</p>

ERROR STRING	MITIGATION
A storage exception occurred while validating data disk {data disk name} with media link {data disk Uri} for VM {VM name} in Cloud Service {Cloud Service name}. Please ensure that the VHD media link is accessible for this virtual machine	This error can happen if the disks of the VM have been deleted or are not accessible anymore. Please make sure the disks for the VM exist.
VM {vm-name} in HostedService {cloud-service-name} contains Disk with MediaLink {vhd-uri} which has blob name {vhd-blob-name} that isn't supported in Azure Resource Manager.	This error occurs when the name of the blob has a "/" in it which isn't supported in Compute Resource Provider currently.
Migration isn't allowed for Deployment {deployment-name} in HostedService {cloud-service-name} as it isn't in the regional scope. Please refer to <a href="https://aka.ms/regionalscope">https://aka.ms/regionalscope</a> for moving this deployment to regional scope.	In 2014, Azure announced that networking resources will move from a cluster level scope to regional scope. See <a href="https://aka.ms/regionalscope">https://aka.ms/regionalscope</a> for more details. This error happens when the deployment being migrated has not had an update operation, which automatically moves it to a regional scope. The best work-around is to either add an endpoint to a VM, or a data disk to the VM, and then retry migration. See <a href="#">How to set up endpoints on a classic virtual machine in Azure</a> or <a href="#">Attach a data disk to a virtual machine created with the classic deployment model</a>
Migration isn't supported for Virtual Network {vnet-name} because it has non-gateway PaaS deployments.	This error occurs when you have non-gateway PaaS deployments such as Application Gateway or API Management services that are connected to the Virtual Network.
Management operations on VM are disallowed because migration is in progress	This error occurs because the VM is in Prepare state and therefore locked for any update/delete operation. Call Abort using PS/CLI on the VM to rollback the migration and unlock the VM for update/delete operations. Calling commit will also unlock the VM but will commit the migration to ARM.

## Detailed mitigations

### **VM with Data Disk whose physical blob size bytes does not match the VM Data Disk logical size bytes.**

This happens when the Data disk logical size can get out of sync with the actual VHD blob size. This can be easily verified using the following commands:

#### **Verifying the issue**

```

# Store the VM details in the VM object
$vm = Get-AzureVM -ServiceName $servicename -Name $vmname

# Display the data disk properties
# NOTE the data disk LogicalDiskSizeInGB below which is 11GB. Also note the MediaLink Uri of the VHD blob as
we'll use this in the next step
$vm.VM.DataVirtualHardDisks


HostCaching      : None
DiskLabel        :
DiskName         : coreosvm-coreosvm-0-201611230636240687
Lun              : 0
LogicalDiskSizeInGB : 11
MediaLink        : https://contosostorage.blob.core.windows.net/vhds/coreosvm-dd1.vhd
SourceMediaLink   :
IOType           : Standard
ExtensionData    :

# Now get the properties of the blob backing the data disk above
# NOTE the size of the blob is about 15 GB which is different from LogicalDiskSizeInGB above
blob = Get-AzStorageblob -Blob "coreosvm-dd1.vhd" -Container vhds

blob

ICloudBlob      : Microsoft.WindowsAzure.Storage.Blob.CloudPageBlob
BlobType        : PageBlob
Length          : 16106127872
ContentType     : application/octet-stream
LastModified    : 11/23/2016 7:16:22 AM +00:00
SnapshotTime    :
ContinuationToken :
Context          : Microsoft.WindowsAzure.Commands.Common.Storage.AzureStorageContext
Name            : coreosvm-dd1.vhd

```

## Mitigating the issue

```

# Convert the blob size in bytes to GB into a variable which we'll use later
$newSize = [int]($blob.Length / 1GB)

# See the calculated size in GB
$newSize

15

# Store the disk name of the data disk as we'll use this to identify the disk to be updated
$diskName = $vm.VM.DataVirtualHardDisks[0].DiskName

# Identify the LUN of the data disk to remove
$lunToRemove = $vm.VM.DataVirtualHardDisks[0].Lun

# Now remove the data disk from the VM so that the disk isn't leased by the VM and it's size can be updated
Remove-AzureDataDisk -LUN $lunToRemove -VM $vm | Update-AzureVm -Name $vmname -ServiceName $servicename

OperationDescription OperationId          OperationStatus
----- ----- -----
Update-AzureVM      213xx1-b44b-1v6n-23gg-591f2a13cd16 Succeeded

# Verify we have the right disk that's going to be updated
Get-AzureDisk -DiskName $diskName

AffinityGroup      :
AttachedTo        :
IsCorrupted      : False
Label             :
Location          : East US
DiskSizeInGB      : 11

```

```

DISKSIZEINGB          : 11
MediaLink              : https://contosostorage.blob.core.windows.net/vhds/coreosvm-dd1.vhd
DiskName               : coreosvm-coreosvm-0-201611230636240687
SourceImageName        :
OS                     :
IOType                 : Standard
OperationDescription   : Get-AzureDisk
OperationId            : 0c56a2b7-a325-123b-7043-74c27d5a61fd
OperationStatus         : Succeeded

# Now update the disk to the new size
Update-AzureDisk -DiskName $diskName -ResizedSizeInGB $newSize -Label $diskName

OperationDescription OperationId          OperationStatus
-----  -----  -----
Update-AzureDisk      cv134b65-1b6n-8908-abuo-ce9e395ac3e7 Succeeded

# Now verify that the "DiskSizeInGB" property of the disk matches the size of the blob
Get-AzureDisk -DiskName $diskName

AffinityGroup          :
AttachedTo              :
IsCorrupted             : False
Label                  : coreosvm-coreosvm-0-201611230636240687
Location                : East US
DiskSizeInGB            : 15
MediaLink               : https://contosostorage.blob.core.windows.net/vhds/coreosvm-dd1.vhd
DiskName                : coreosvm-coreosvm-0-201611230636240687
SourceImageName         :
OS                     :
IOType                 : Standard
OperationDescription   : Get-AzureDisk
OperationId            : 1v53bde5-cv56-5621-9078-16b9c8a0bad2
OperationStatus         : Succeeded

# Now we'll add the disk back to the VM as a data disk. First we need to get an updated VM object
$vm = Get-AzureVM -ServiceName $servicename -Name $vmname

Add-AzureDataDisk -Import -DiskName $diskName -LUN 0 -VM $vm -HostCaching ReadWrite | Update-AzureVm -Name
$vmname -ServiceName $servicename

OperationDescription OperationId          OperationStatus
-----  -----  -----
Update-AzureVM       b0ad3d4c-4v68-45vb-xxc1-134fd010d0f8 Succeeded

```

## Moving a VM to a different subscription after completing migration

After you complete the migration process, you may want to move the VM to another subscription. However, if you have a secret/certificate on the VM that references a Key Vault resource, the move is currently not supported. The below instructions will allow you to work around the issue.

### PowerShell

```

$vm = Get-AzVM -ResourceGroupName "MyRG" -Name "MyVM"
Remove-AzVMSecret -VM $vm
Update-AzVM -ResourceGroupName "MyRG" -VM $vm

```

### Azure CLI

```
az vm update -g "myrg" -n "myvm" --set osProfile.Secrets=[]
```

## Next steps

- Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager
- Technical deep dive on platform-supported migration from classic to Azure Resource Manager
- Planning for migration of IaaS resources from classic to Azure Resource Manager
- Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager
- Use CLI to migrate IaaS resources from classic to Azure Resource Manager
- Community tools for assisting with migration of IaaS resources from classic to Azure Resource Manager
- Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager

# Community tools to migrate IaaS resources from classic to Azure Resource Manager

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## IMPORTANT

Today, about 90% of IaaS VMs are using [Azure Resource Manager](#). As of February 28, 2020, classic VMs have been deprecated and will be fully retired on March 1, 2023. [Learn more](#) about this deprecation and [how it affects you](#).

This article catalogs the tools that have been provided by the community to assist with migration of IaaS resources from classic to the Azure Resource Manager deployment model.

## NOTE

These tools are not officially supported by Microsoft Support. Therefore they are open sourced on GitHub and we're happy to accept PRs for fixes or additional scenarios. To report an issue, use the GitHub issues feature.

Migrating with these tools will cause downtime for your classic Virtual Machine. If you're looking for platform supported migration, visit

- [Platform supported migration of IaaS resources from Classic to Azure Resource Manager stack](#)
- [Technical Deep Dive on Platform supported migration from Classic to Azure Resource Manager](#)
- [Migrate IaaS resources from Classic to Azure Resource Manager using Azure PowerShell](#)

## AsmMetadataParser

This is a collection of helper tools created as part of enterprise migrations from Azure Service Management to Azure Resource Manager. This tool allows you to replicate your infrastructure into another subscription which can be used for testing migration and iron out any issues before running the migration on your Production subscription.

[Link to the tool documentation](#)

## migAz

migAz is an additional option to migrate a complete set of classic IaaS resources to Azure Resource Manager IaaS resources. The migration can occur within the same subscription or between different subscriptions and subscription types (ex: CSP subscriptions).

- [Link to the tool documentation](#)

## Next Steps

- [Overview of platform-supported migration of IaaS resources from classic to Azure Resource Manager](#)
- [Technical deep dive on platform-supported migration from classic to Azure Resource Manager](#)
- [Planning for migration of IaaS resources from classic to Azure Resource Manager](#)
- [Use PowerShell to migrate IaaS resources from classic to Azure Resource Manager](#)

- Use CLI to migrate IaaS resources from classic to Azure Resource Manager
- Review most common migration errors
- Review the most frequently asked questions about migrating IaaS resources from classic to Azure Resource Manager



# Availability options for Azure Virtual Machines

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article provides an overview of the availability options for Azure virtual machines (VMs).

## Availability zones

[Availability zones](#) expands the level of control you have to maintain the availability of the applications and data on your VMs. An Availability Zone is a physically separate zone, within an Azure region. There are three Availability Zones per supported Azure region.

Each Availability Zone has a distinct power source, network, and cooling. By designing your solutions to use replicated VMs in zones, you can protect your apps and data from the loss of a data center. If one zone is compromised, then replicated apps and data are instantly available in another zone.

## Virtual Machines Scale Sets

[Azure virtual machine scale sets](#) let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update many VMs. There is no cost for the scale set itself, you only pay for each VM instance that you create.

Virtual machines in a scale set can also be deployed into multiple availability zones, a single availability zone, or regionally. Availability zone deployment options may differ based on the [orchestration mode](#).

## Availability sets

An [availability set](#) is a logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability. We recommend that two or more VMs are created within an availability set to provide for a highly available application and to meet the [99.95% Azure SLA](#). There is no cost for the Availability Set itself, you only pay for each VM instance that you create.

## Load balancer

Combine the [Azure Load Balancer](#) with an availability zone or availability set to get the most application resiliency. The Azure Load Balancer distributes traffic between multiple virtual machines. For our Standard tier virtual machines, the Azure Load Balancer is included. Not all virtual machine tiers include the Azure Load Balancer. For more information about load balancing your virtual machines, see [Load Balancing virtual machines](#) for [Linux](#) or [Windows](#).

## Azure Storage redundancy

Azure Storage always stores multiple copies of your data so that it is protected from planned and unplanned events, including transient hardware failures, network or power outages, and massive natural disasters. Redundancy ensures that your storage account meets its availability and durability targets even in the face of failures.

When deciding which redundancy option is best for your scenario, consider the tradeoffs between lower costs and higher availability. The factors that help determine which redundancy option you should choose include:

- How your data is replicated in the primary region
- Whether your data is replicated to a second region that is geographically distant to the primary region, to protect against regional disasters
- Whether your application requires read access to the replicated data in the secondary region if the primary region becomes unavailable for any reason

For more information, see [Azure Storage redundancy](#)

## Azure Site Recovery

As an organization you need to adopt a business continuity and disaster recovery (BCDR) strategy that keeps your data safe, and your apps and workloads online, when planned and unplanned outages occur.

[Azure Site Recovery](#) helps ensure business continuity by keeping business apps and workloads running during outages. Site Recovery replicates workloads running on physical and virtual machines (VMs) from a primary site to a secondary location. When an outage occurs at your primary site, you fail over to secondary location, and access apps from there. After the primary location is running again, you can fail back to it.

Site Recovery can manage replication for:

- Azure VMs replicating between Azure regions.
- On-premises VMs, Azure Stack VMs, and physical servers.

## Next steps

- [Create a virtual machine in an availability zone](#)
- [Create a virtual machine in an availability set](#)
- [Create a virtual machine scale set](#)

# Regions and availability zones

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure regions and availability zones are designed to help you achieve resiliency and reliability for your business-critical workloads. Azure maintains multiple geographies. These discrete demarcations define disaster recovery and data residency boundaries across one or multiple Azure regions. Maintaining many regions ensures customers are supported across the world.

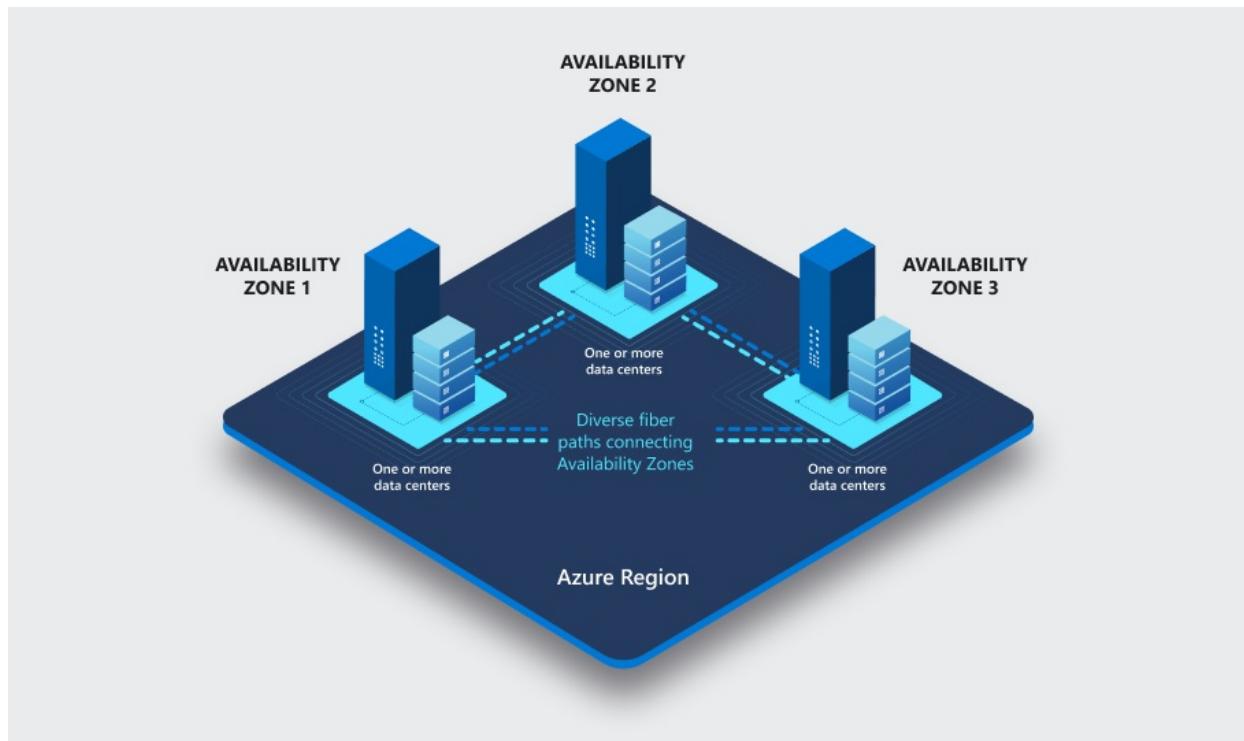
## Regions

Each Azure region features datacenters deployed within a latency-defined perimeter. They're connected through a dedicated regional low-latency network. This design ensures that Azure services within any region offer the best possible performance and security.

## Availability zones

Azure *availability zones* are physically separate locations within each Azure region that are tolerant to local failures. Failures can range from software and hardware failures to events such as earthquakes, floods, and fires. Tolerance to failures is achieved because of redundancy and logical isolation of Azure services. To ensure resiliency, a minimum of three separate availability zones are present in all availability zone-enabled regions.

Azure availability zones are connected by a high-performance network with a round-trip latency of less than 2ms. They help your data stay synchronized and accessible when things go wrong. Each zone is composed of one or more datacenters equipped with independent power, cooling, and networking infrastructure. Availability zones are designed so that if one zone is affected, regional services, capacity, and high availability are supported by the remaining two zones.



Datacenter locations are selected by using rigorous vulnerability risk assessment criteria. This process identifies all significant datacenter-specific risks and considers shared risks between availability zones.

With availability zones, you can design and operate applications and databases that automatically transition

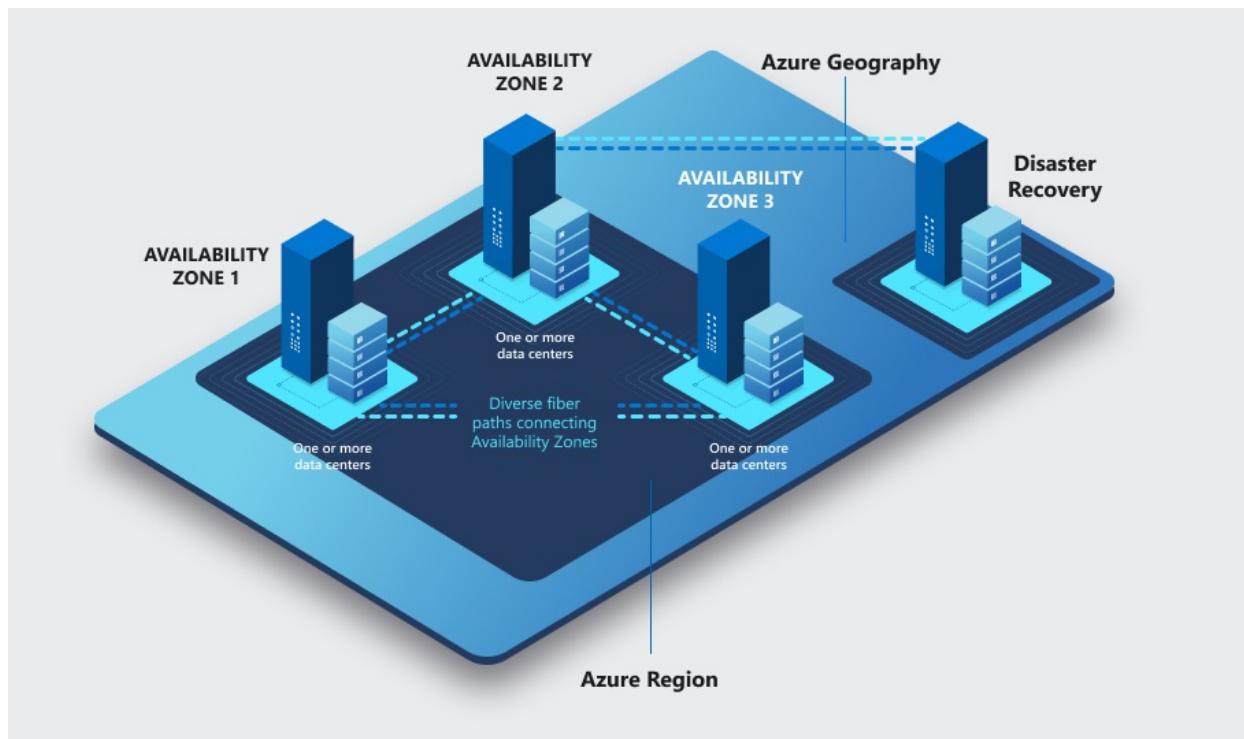
between zones without interruption. Azure availability zones are highly available, fault tolerant, and more scalable than traditional single or multiple datacenter infrastructures.

Each data center is assigned to a physical zone. Physical zones are mapped to logical zones in your Azure subscription. Azure subscriptions are automatically assigned this mapping at the time a subscription is created. You can use the dedicated ARM API called: [checkZonePeers](#) to compare zone mapping for resilient solutions that span across multiple subscriptions.

You can design resilient solutions by using Azure services that use availability zones. Co-locate your compute, storage, networking, and data resources across an availability zone, and replicate this arrangement in other availability zones.

Azure *availability zones-enabled services* are designed to provide the right level of resiliency and flexibility. They can be configured in two ways. They can be either *zone redundant*, with automatic replication across zones, or *zonal*, with instances pinned to a specific zone. You can also combine these approaches.

Some organizations require high availability of availability zones and protection from large-scale phenomena and regional disasters. Azure regions are designed to offer protection against localized disasters with availability zones and protection from regional or large geography disasters with disaster recovery, by making use of another region. To learn more about business continuity, disaster recovery, and cross-region replication, see [Cross-region replication in Azure](#).



## Azure regions with availability zones

Azure provides the most extensive global footprint of any cloud provider and is rapidly opening new regions and availability zones. Azure has availability zones in every country in which Azure operates a datacenter region. The following regions currently support availability zones.

AMERICAS	EUROPE	MIDDLE EAST	AFRICA	ASIA PACIFIC
Brazil South	France Central	Qatar Central	South Africa North	Australia East
Canada Central	Germany West Central	UAE North		Central India

AMERICAS	EUROPE	MIDDLE EAST	AFRICA	ASIA PACIFIC
Central US	North Europe			Japan East
East US	Norway East			Korea Central
East US 2	UK South			Southeast Asia
South Central US	West Europe			East Asia
US Gov Virginia	Sweden Central			China North 3
West US 2	Switzerland North			
West US 3				

## Next steps

- Microsoft commitment to expand Azure availability zones to more regions
- Azure services that support availability zones
- Azure services

# Create a virtual machine in an availability zone using Azure CLI

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article steps through using the Azure CLI to create a Linux VM in an Azure availability zone. An [availability zone](#) is a physically separate zone in an Azure region. Use availability zones to protect your apps and data from an unlikely failure or loss of an entire datacenter.

To use an availability zone, create your virtual machine in a [supported Azure region](#).

Make sure that you have installed the latest [Azure CLI](#) and logged in to an Azure account with [az login](#).

## Check VM SKU availability

The availability of VM sizes, or SKUs, may vary by region and zone. To help you plan for the use of Availability Zones, you can list the available VM SKUs by Azure region and zone. This ability makes sure that you choose an appropriate VM size, and obtain the desired resiliency across zones. For more information on the different VM types and sizes, see [VM Sizes overview](#).

You can view the available VM SKUs with the `az vm list-skus` command. The following example lists available VM SKUs in the `eastus2` region:

```
az vm list-skus --location eastus2 --output table
```

The output is similar to the following condensed example, which shows the Availability Zones in which each VM size is available:

ResourceType	Locations	Name	[...]	Tier	Size	Zones
virtualMachines	eastus2	Standard_DS1_v2		Standard	DS1_v2	1,2,3
virtualMachines	eastus2	Standard_DS2_v2		Standard	DS2_v2	1,2,3
[...]						
virtualMachines	eastus2	Standard_F1s		Standard	F1s	1,2,3
virtualMachines	eastus2	Standard_F2s		Standard	F2s	1,2,3
[...]						
virtualMachines	eastus2	Standard_D2s_v3		Standard	D2_v3	1,2,3
virtualMachines	eastus2	Standard_D4s_v3		Standard	D4_v3	1,2,3
[...]						
virtualMachines	eastus2	Standard_E2_v3		Standard	E2_v3	1,2,3
virtualMachines	eastus2	Standard_E4_v3		Standard	E4_v3	1,2,3

## Create resource group

Create a resource group with the `az group create` command.

An Azure resource group is a logical container into which Azure resources are deployed and managed. A resource group must be created before a virtual machine. In this example, a resource group named `myResourceGroupVM` is created in the `eastus2` region. East US 2 is one of the Azure regions that supports availability zones.

```
az group create --name myResourceGroupVM --location eastus2
```

The resource group is specified when creating or modifying a VM, which can be seen throughout this article.

## Create virtual machine

Create a virtual machine with the [az vm create](#) command.

When creating a virtual machine, several options are available such as operating system image, disk sizing, and administrative credentials. In this example, a virtual machine is created with a name of *myVM* running Ubuntu Server. The VM is created in availability zone 1. By default, the VM is created in the *Standard\_DS1\_v2* size.

```
az vm create --resource-group myResourceGroupVM --name myVM --location eastus2 --image UbuntuLTS --generate-ssh-keys --zone 1
```

It may take a few minutes to create the VM. Once the VM has been created, the Azure CLI outputs information about the VM. Take note of the `zones` value, which indicates the availability zone in which the VM is running.

```
{
  "fqdns": "",
  "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxx/resourceGroups/myResourceGroupVM/providers/Microsoft.Compute/virtualMachines/myVM",
  "location": "eastus2",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "52.174.34.95",
  "resourceGroup": "myResourceGroupVM",
  "zones": "1"
}
```

## Confirm zone for managed disk and IP address

When the VM is deployed in an availability zone, a managed disk for the VM is created in the same availability zone. By default, a public IP address is also created in that zone. The following examples get information about these resources.

To verify that the VM's managed disk is in the availability zone, use the [az vm show](#) command to return the disk ID. In this example, the disk ID is stored in a variable that is used in a later step.

```
osdiskname=$(az vm show -g myResourceGroupVM -n myVM --query "storageProfile.osDisk.name" -o tsv)
```

Now you can get information about the managed disk:

```
az disk show --resource-group myResourceGroupVM --name $osdiskname
```

The output shows that the managed disk is in the same availability zone as the VM:

```
{
  "creationData": {
    "createOption": "FromImage",
    "imageReference": {
      "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/Providers/Microsoft.Compute/Locations/westeurope/Publishers/Canonical/ArtifactTypes/VMImage/Off
ers/UbuntuServer/Skus/16.04-LTS/Versions/latest",
      "lun": null
    },
    "sourceResourceId": null,
    "sourceUri": null,
    "storageAccountId": null
  },
  "diskSizeGb": 30,
  "encryptionSettings": null,
  "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/resourceGroups/myResourceGroupVM/providers/Microsoft.Compute/disks/osdisk_761c570dab",
  "location": "eastus2",
  "managedBy": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/resourceGroups/myResourceGroupVM/providers/Microsoft.Compute/virtualMachines/myVM",
  "name": "myVM_osdisk_761c570dab",
  "osType": "Linux",
  "provisioningState": "Succeeded",
  "resourceGroup": "myResourceGroupVM",
  "sku": {
    "name": "Premium_LRS",
    "tier": "Premium"
  },
  "tags": {},
  "timeCreated": "2018-03-05T22:16:06.892752+00:00",
  "type": "Microsoft.Compute/disks",
  "zones": [
    "1"
  ]
}
```

Use the [az vm list-ip-addresses](#) command to return the name of public IP address resource in *myVM*. In this example, the name is stored in a variable that is used in a later step.

```
ipaddressname=$(az vm list-ip-addresses -g myResourceGroupVM -n myVM --query "
[].virtualMachine.network.publicIpAddresses[].name" -o tsv)
```

Now you can get information about the IP address:

```
az network public-ip show --resource-group myResourceGroupVM --name $ipaddressname
```

The output shows that the IP address is in the same availability zone as the VM:

```
{  
  "dnsSettings": null,  
  "etag": "W/\"b7ad25eb-3191-4c8f-9cec-c5e4a3a37d35\"",  
  "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxxxx/resourceGroups/myResourceGroupVM/providers/Microsoft.Network/publicIPAddresses/myVMPublicIP",  
  "idleTimeoutInMinutes": 4,  
  "ipAddress": "52.174.34.95",  
  "ipConfiguration": {  
    "etag": null,  
    "id": "/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-  
xxxxxxxxxxxx/resourceGroups/myResourceGroupVM/providers/Microsoft.Network/networkInterfaces/myVMVMNic/ipConf  
igurations/ipconfigmyVM",  
    "name": null,  
    "privateIpAddress": null,  
    "privateIpAllocationMethod": null,  
    "provisioningState": null,  
    "publicIpAddress": null,  
    "resourceGroup": "myResourceGroupVM",  
    "subnet": null  
  },  
  "location": "eastUS2",  
  "name": "myVMPublicIP",  
  "provisioningState": "Succeeded",  
  "publicIpAddressVersion": "IPv4",  
  "publicIpAllocationMethod": "Dynamic",  
  "resourceGroup": "myResourceGroupVM",  
  "resourceGuid": "8c70a073-09be-4504-0000-000000000000",  
  "tags": {},  
  "type": "Microsoft.Network/publicIPAddresses",  
  "zones": [  
    "1"  
  ]  
}
```

## Next steps

In this article, you learned how to create a VM in an availability zone. Learn more about [availability](#) for Azure VMs.

# Create a virtual machine in an availability zone using Azure PowerShell

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article details using Azure PowerShell to create an Azure virtual machine running Windows Server 2016 in an Azure availability zone. An [availability zone](#) is a physically separate zone in an Azure region. Use availability zones to protect your apps and data from an unlikely failure or loss of an entire datacenter.

To use an availability zone, create your virtual machine in a [supported Azure region](#).

## Sign in to Azure

Sign in to your Azure subscription with the `Connect-AzAccount` command and follow the on-screen directions.

```
Connect-AzAccount
```

## Check VM SKU availability

The availability of VM sizes, or SKUs, may vary by region and zone. To help you plan for the use of Availability Zones, you can list the available VM SKUs by Azure region and zone. This ability makes sure that you choose an appropriate VM size, and obtain the desired resiliency across zones. For more information on the different VM types and sizes, see [VM Sizes overview](#).

You can view the available VM SKUs with the `Get-AzComputeResourceSku` command. The following example lists available VM SKUs in the *eastus2* region:

```
Get-AzComputeResourceSku | where {$_.Locations.Contains("eastus2")};
```

The output is similar to the following condensed example, which shows the Availability Zones in which each VM size is available:

ResourceType	Name	Location	Zones	[...]
-----	---	-----	-----	-----
virtualMachines	Standard_DS1_v2	eastus2	{1, 2, 3}	
virtualMachines	Standard_DS2_v2	eastus2	{1, 2, 3}	
[...]				
virtualMachines	Standard_F1s	eastus2	{1, 2, 3}	
virtualMachines	Standard_F2s	eastus2	{1, 2, 3}	
[...]				
virtualMachines	Standard_D2s_v3	eastus2	{1, 2, 3}	
virtualMachines	Standard_D4s_v3	eastus2	{1, 2, 3}	
[...]				
virtualMachines	Standard_E2_v3	eastus2	{1, 2, 3}	
virtualMachines	Standard_E4_v3	eastus2	{1, 2, 3}	

## Create resource group

Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which

Azure resources are deployed and managed. In this example, a resource group named *myResourceGroup* is created in the *eastus2* region.

```
New-AzResourceGroup -Name myResourceGroup -Location EastUS2
```

## Create networking resources

### Create a virtual network, subnet, and a public IP address

These resources are used to provide network connectivity to the virtual machine and connect it to the internet. Create the IP address in an availability zone, 2 in this example. In a later step, you create the VM in the same zone used to create the IP address.

```
# Create a subnet configuration
$subnetConfig = New-AzVirtualNetworkSubnetConfig -Name mySubnet -AddressPrefix 192.168.1.0/24

# Create a virtual network
$vnet = New-AzVirtualNetwork -ResourceGroupName myResourceGroup -Location eastus2 ` 
    -Name myVNet -AddressPrefix 192.168.0.0/16 -Subnet $subnetConfig

# Create a public IP address in an availability zone and specify a DNS name
$pip = New-AzPublicIpAddress -ResourceGroupName myResourceGroup -Location eastus2 -Zone 2 ` 
    -AllocationMethod Static -IdleTimeoutInMinutes 4 -Name "mypublicdns$(Get-Random)" -Sku Standard
```

### Create a network security group and a network security group rule

The network security group secures the virtual machine using inbound and outbound rules. In this case, an inbound rule is created for port 3389, which allows incoming remote desktop connections. We also want to create an inbound rule for port 80, which allows incoming web traffic.

```
# Create an inbound network security group rule for port 3389 - change -Access to "Allow" if you want to
allow RDP access
$nsgRuleRDP = New-AzNetworkSecurityRuleConfig -Name myNetworkSecurityGroupRuleRDP -Protocol Tcp ` 
    -Direction Inbound -Priority 1000 -SourceAddressPrefix * -SourcePortRange * -DestinationAddressPrefix * ` 
    -DestinationPortRange 3389 -Access Deny

# Create an inbound network security group rule for port 80 - - change -Access to "Allow" if you want to
allow TCP traffic over port 80
$nsgRuleWeb = New-AzNetworkSecurityRuleConfig -Name myNetworkSecurityGroupRuleWWW -Protocol Tcp ` 
    -Direction Inbound -Priority 1001 -SourceAddressPrefix * -SourcePortRange * -DestinationAddressPrefix * ` 
    -DestinationPortRange 80 -Access Deny

# Create a network security group
$nsg = New-AzNetworkSecurityGroup -ResourceGroupName myResourceGroup -Location eastus2 ` 
    -Name myNetworkSecurityGroup -SecurityRules $nsgRuleRDP,$nsgRuleWeb
```

### Create a network card for the virtual machine

Create a network card with [New-AzNetworkInterface](#) for the virtual machine. The network card connects the virtual machine to a subnet, network security group, and public IP address.

```
# Create a virtual network card and associate with public IP address and NSG
$nic = New-AzNetworkInterface -Name myNic -ResourceGroupName myResourceGroup -Location eastus2 ` 
    -SubnetId $vnet.Subnets[0].Id -PublicIpAddressId $pip.Id -NetworkSecurityGroupId $nsg.Id
```

## Create virtual machine

Create a virtual machine configuration. This configuration includes the settings that are used when deploying the virtual machine such as a virtual machine image, size, and authentication configuration. The *Standard\_DS1\_v2* size in this example is supported in availability zones. This configuration also specifies the availability zone you set when creating the IP address. When running this step, you are prompted for credentials. The values that you enter are configured as the user name and password for the virtual machine.

```
# Define a credential object
$cred = Get-Credential

# Create a virtual machine configuration
$vmConfig = New-AzVMConfig -VMName myVM -VMSize Standard_DS1_v2 -Zone 2 | ` 
    Set-AzVMOperatingSystem -Windows -ComputerName myVM -Credential $cred | ` 
    Set-AzVMSourceImage -PublisherName MicrosoftWindowsServer -Offer WindowsServer ` 
    -Sku 2016-Datacenter -Version latest | Add-AzVMNetworkInterface -Id $nic.Id
```

Create the virtual machine with [New-AzVM](#).

```
New-AzVM -ResourceGroupName myResourceGroup -Location eastus2 -VM $vmConfig
```

## Confirm zone for managed disk

You created the VM's IP address resource in the same availability zone as the VM. The managed disk resource for the VM is created in the same availability zone. You can verify this with [Get-AzDisk](#):

```
Get-AzDisk -ResourceGroupName myResourceGroup
```

The output shows that the managed disk is in the same availability zone as the VM:

```
ResourceGroupName : myResourceGroup
AccountType      : PremiumLRS
OwnerId          : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/resourceGroups/myResourceGroup/providers/Microsoft. 
                    Compute/virtualMachines/myVM
ManagedBy        : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx//resourceGroups/myResourceGroup/providers/Microsoft. 
                    Compute/virtualMachines/myVM
Sku              : Microsoft.Azure.Management.Compute.Models.DiskSku
Zones            : {2}
TimeCreated      : 9/7/2017 6:57:26 PM
OsType           : Windows
CreationData     : Microsoft.Azure.Management.Compute.Models.CreationData
DiskSizeGB       : 127
EncryptionSettings :
ProvisioningState : Succeeded
Id               : /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/resourceGroups/myResourceGroup/providers/Microsoft. 
                    Compute/disks/myVM_OsDisk_1_bd921920bb0a4650becfc2d830000000
Name             : myVM_OsDisk_1_bd921920bb0a4650becfc2d830000000
Type             : Microsoft.Compute/disks
Location         : eastus2
Tags             : {}
```

## Next steps

In this article, you learned how to create a VM in an availability zone. Learn more about [availability](#) for Azure VMs.

# Create virtual machines in an availability zone using the Azure portal

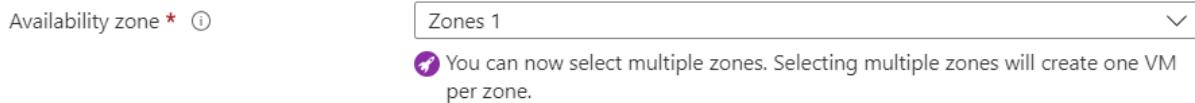
9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

This article steps through using the Azure portal to create highly resilient virtual machines in [availability zones](#). Azure availability zones are physically separate locations within each Azure region that are tolerant to local failures. Use availability zones to protect your applications and data against unlikely datacenter failures.

To use availability zones, create your virtual machines in a [supported Azure region](#).

Some users will now see the option to create VMs in multiple zones. If you see the following message, use the **Preview** tab below.



- [Standard](#)
- [Preview](#)

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Click **Create a resource > Compute > Virtual machine**.
3. Enter the virtual machine information. The user name and password or SSH key is used to sign in to the virtual machine.
4. Choose a region such as East US 2 that supports availability zones.
5. Under **Availability options**, select **Availability zone** dropdown.
6. Under **Availability zone**, select a zone from the drop-down list.
7. Choose a size for the VM. Select a recommended size, or filter based on features. Confirm the size is available in the zone you want to use.
8. Finish filling in the information for your VM. When you are done, select **Review + create**.
9. Once the information is verified, select **Create**.
10. After the VM is created, you can see the availability zone listed in the **Essentials section** on the page for the VM.

## Next steps

In this article, you learned how to create a VM in an availability zone. Learn more about [availability](#) for Azure VMs.

# Migrate Virtual Machines and Virtual Machine Scale Sets to availability zone support

9/21/2022 • 6 minutes to read • [Edit Online](#)

This guide describes how to migrate Virtual Machines (VMs) and Virtual Machine Scale Sets (VMSS) from non-availability zone support to availability zone support. We'll take you through the different options for migration, including how you can use availability zone support for Disaster Recovery solutions.

Virtual Machine (VM) and Virtual Machine Scale Sets (VMSS) are zonal services, which means that VM resources can be deployed by using one of the following methods:

- VM resources are deployed to a specific, self-selected availability zone to achieve more stringent latency or performance requirements.
- VM resources are replicated to one or more zones within the region to improve the resiliency of the application and data in a High Availability (HA) architecture.

When you migrate resources to availability zone support, we recommend that you select multiple zones for your new VMs and VMSS, to ensure high-availability of your compute resources.

## Prerequisites

To migrate to availability zone support, your VM SKUs must be available across the zones in for your region. To check for VM SKU availability, use one of the following methods:

- Use PowerShell to [Check VM SKU availability](#).
- Use the Azure CLI to [Check VM SKU availability](#).
- Go to [Foundational Services](#).

## Downtime requirements

Because zonal VMs are created across the availability zones, all migration options mentioned in this article require downtime during deployment.

## Migration Option 1: Redeployment

### When to use redeployment

Use the redeployment option if you have set up good Infrastructure as Code (IaC) practices to manage infrastructure. This redeployment option gives you more control and the ability to automate various processes within your deployment pipelines.

### Redeployment considerations

- When you redeploy your VM and VMSS resources, the underlying resources such as managed disk and IP address for the VM are created in the same availability zone. You must use a Standard SKU public IP address and load balancer to create zone-redundant network resources.
- Existing managed disks without availability zone support can't be attached to a VM with availability zone support. To attach existing managed disks to a VM with availability zone support, you'll need to take a snapshot of the current disks, and then create your VM with the new managed disks attached.
- For zonal deployments that require reasonably low network latency and good performance between

application tier and data tier, use [proximity placement groups](#). Proximity groups can force grouping of different VM resources under a single network spine. For an example of an SAP workload that uses proximity placement groups, see [Azure proximity placement groups for optimal network latency with SAP applications](#)

## How to redeploy

If you want to migrate the data on your current managed disks when creating a new VM, follow the directions in [Migrate your managed disks](#).

If you only want to create new VM with new managed disks in an availability zone, see:

- [Create VM using Azure CLI](#)
- [Create VM using Azure PowerShell](#)
- [Create VM using Azure portal](#)

To learn how to create VMSS in an availability zone, see [Create a virtual machine scale set that uses Availability Zones](#).

## Migrate your managed disks

In this section, you'll migrate the data from your current managed disks to either zone-redundant storage (ZRS) managed disks or zonal managed disks.

### Step 1: Create your snapshot

The easiest and cleanest way to create a snapshot is to do so while the VM is offline. See [Create snapshots while the VM is offline](#). If you choose this approach, some downtime should be expected. To create a snapshot of your VM using the Azure portal, PowerShell, or Azure CLI, see [Create a snapshot of a virtual hard disk](#)

If you'll be taking a snapshot of a disk that's attached to a running VM, read the guidance in [Create snapshots while the VM is running](#) before proceeding.

#### NOTE

The source managed disks remain intact with their current configurations and you'll continue to be billed for them. To avoid this, you must manually delete the disks once you've finished your migration and confirmed the new disks are working. For more information, see [Find and delete unattached Azure managed and unmanaged disks](#).

### Step 2: Migrate the data on your managed disks

Now that you have snapshots of your original disks, you can use them to create either ZRS managed disks or zonal managed disks.

#### Migrate your data to zonal managed disks

To migrate a non-zonal managed disk to zonal:

1. Create a zonal managed disk from the source disk snapshot. The zone parameter should match your zonal VM. To create a zonal managed disk from the snapshot, you can use [Azure CLI](#)(example below), [PowerShell](#), or the [Azure Portal](#).

```
az disk create --resource-group $resourceGroupName --name $diskName --location $location --zone  
$zone --sku $storageType --size-gb $diskSize --source $snapshotId
```

#### Migrate your data to ZRS managed disks

#### IMPORTANT

Zone-redundant storage (ZRS) for managed disks has some restrictions. For more information see [Limitations](#).

1. Create a ZRS managed disk from the source disk snapshot by using the following Azure CLI snippet:

```
# Create a new ZRS Managed Disks using the snapshot Id and the SKU supported  
storageType=Premium_ZRS  
location=westus2  
  
az disk create --resource-group $resourceGroupName --name $diskName --sku $storageType --size-gb  
$diskSize --source $snapshotId
```

### Step 3: Create a new VM with your new disks

Now that you have migrated your data to ZRS managed disks or zonal managed disks, create a new VM with these new disks set as the OS and data disks:

```
az vm create -g MyResourceGroup -n MyVm --attach-os-disk newZonalOSDiskCopy --attach-data-disks  
newZonalDataDiskCopy --os-type linux
```

## Migration Option 2: Azure Resource Mover

### When to use Azure Resource Mover

Use Azure Resource Mover for an easy way to move VMs or encrypted VMs from one region without availability zones to another with availability zone support. If you want to learn more about the benefits of using Azure Resource Mover, see [Why use Azure Resource Mover?](#).

### Azure Resource Mover considerations

When you use Azure Resource mover, all keys and secrets are copied from the source key vault to the newly created destination key vault in your target region. All resources related to your customer-managed keys, such as Azure Key Vaults, disk encryption sets, VMs, disks, and snapshots, must be in the same subscription and region. Azure Key Vault's default availability and redundancy feature can't be used as the destination key vault for the moved VM resources, even if the target region is a secondary region to which your source key vault is replicated.

### How to use Azure Resource Mover

To learn how to move VMs to another region, see [Move Azure VMs to an availability zone in another region](#)

To learn how to move encrypted VMs to another region, see [Tutorial: Move encrypted Azure VMs across regions](#)

## Disaster Recovery Considerations

Typically, availability zones are used to deploy VMs in a High Availability configuration. They may be too close to each other to serve as a Disaster Recovery solution during a natural disaster. However, there are scenarios where availability zones can be used for Disaster Recovery. To learn more, see [Using Availability Zones for Disaster Recovery](#).

The following requirements should be part of a disaster recovery strategy that helps your organization run its workloads during planned or unplanned outages across zones:

- The source VM must already be a zonal VM, which means that it's placed in a logical zone.
- You'll need to replicate your VM from one zone to another zone using Azure Site Recovery service.
- Once your VM is replicated to another zone, you can follow steps to run a Disaster Recovery drill, fail over, reprotect, and failback.
- To enable VM disaster recovery between availability zones, follow the instructions in [Enable Azure VM](#)

disaster recovery between availability zones .

## Next Steps

Learn more about:

[Regions and Availability Zones in Azure](#)

[Azure Services that support Availability Zones](#)

# Proximity placement groups

9/21/2022 • 9 minutes to read • [Edit Online](#)

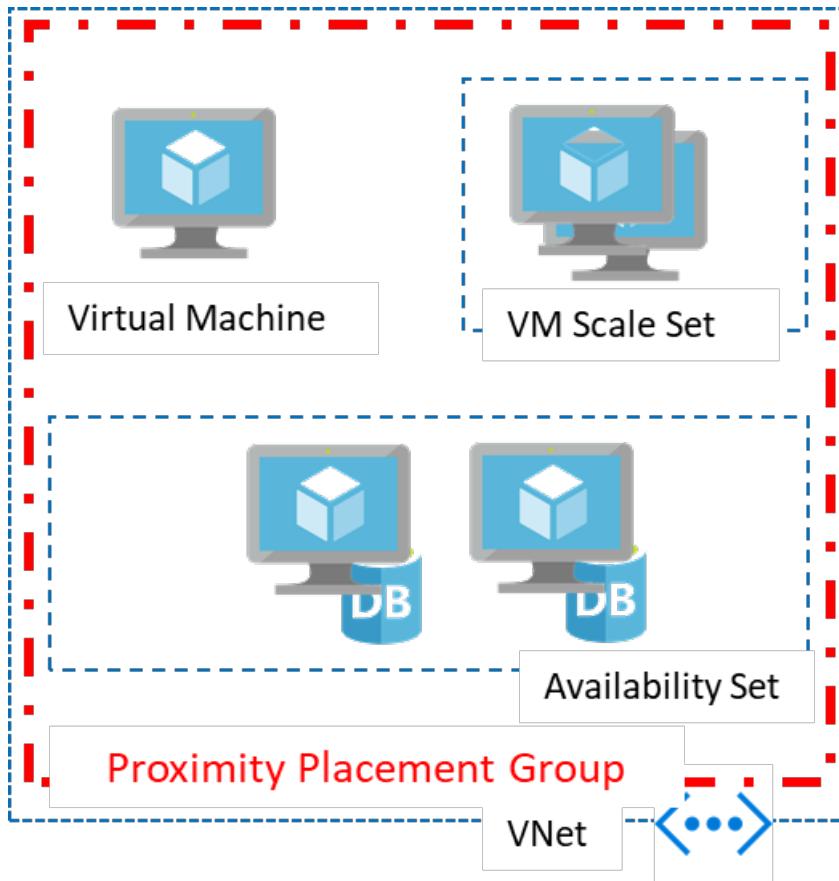
Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Placing VMs in a single region reduces the physical distance between the instances. Placing them within a single availability zone will also bring them physically closer together. However, as the Azure footprint grows, a single availability zone may span multiple physical data centers, which may result in a network latency impacting your application.

To get VMs as close as possible, achieving the lowest possible latency, you should deploy them within a proximity placement group.

A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

- Low latency between stand-alone VMs.
- Low Latency between VMs in a single availability set or a virtual machine scale set.
- Low latency between stand-alone VMs, VMs in multiple Availability Sets, or multiple scale sets. You can have multiple compute resources in a single placement group to bring together a multi-tiered application.
- Low latency between multiple application tiers using different hardware types. For example, running the backend using M-series in an availability set and the front end on a D-series instance, in a scale set, in a single proximity placement group.



## Using Proximity Placement Groups

A proximity placement group is a resource in Azure. You need to create one before using it with other resources. Once created, it could be used with virtual machines, availability sets, or virtual machine scale sets. You specify a proximity placement group when creating compute resources providing the proximity placement group ID.

You can also move an existing resource into a proximity placement group. When moving a resource into a proximity placement group, you should stop (deallocate) the asset first since it will be redeployed potentially into a different data center in the region to satisfy the colocation constraint.

In the case of availability sets and virtual machine scale sets, you should set the proximity placement group at the resource level rather than the individual virtual machines.

A proximity placement group is a colocation constraint rather than a pinning mechanism. It's pinned to a specific data center with the deployment of the first resource to use it. Once all resources using the proximity placement group have been stopped (deallocated) or deleted, it's no longer pinned. Therefore, whenever you use a proximity placement group with multiple VM series, it's important to specify all the required types upfront in a template if possible or follow a deployment sequence, which will improve your chances for a successful deployment. If your deployment fails, restart the deployment with the VM size, which has failed as the first size to be deployed.

## Use intent to specify VM sizes

You can use the optional `intent` parameter to provide the intended [VM Sizes](#) to be part of the proximity placement group. This parameter can be specified at the time of creating a proximity placement group or it can be added/modified while updating a proximity placement group after deallocating all of the VMs.

When specifying `intent`, you can also add the optional `zone` parameter to specify an availability zone, indicating that the proximity placement group must be created within a specific availability zone. Note the following points when providing the `zone` parameter:

- The availability zone parameter can only be provided during the creation of the proximity placement group and can't be modified later.
- The `zone` parameter can only be used with `intent`, it can't be used alone.
- Only one availability zone can be specified.

Proximity Placement Group creation or update will succeed only when at least one data center supports all the VM Sizes specified in the intent. Otherwise, the creation or update will fail with "OverconstrainedAllocationRequest", indicating that the combination of VM Sizes can't be supported within a proximity placement group. The **intent does not provide any capacity reservation or guarantee**. The VM Sizes and zone given in `intent` are used to select an appropriate data center, reducing the chances of failure if the desired VM size isn't available in a data center. Allocation failures can still occur if there is no more capacity for a VM size at the time of deployment.

### NOTE

To use intent for your proximity placement groups, ensure that the API version is 2021-11-01 or higher

### Best Practices while using intent

- Provide an availability zone for your proximity placement group only when you provide an intent. Providing an availability zone without an intent will result in an error when creating the proximity placement group.
- If you provide an availability zone in the intent, ensure that the availability zone of the VMs you deploy match with what is specified in the intent, to avoid errors while deploying VMs.
- Creating or adding VMs with sizes that are not included in the intent is allowed, but not recommended. The size may not exist in the selected datacenter and can result in failures at the time of VM deployment.
- For existing placement groups, we recommend you include the sizes of the existing VMs when updating the

intent, in order to avoid failure when redeploying the VMs.

### Intent can be affected with decommissioning

- It is possible that after creating a proximity placement group with intent and before deploying VMs, planned maintenance events such as hardware decommissioning at an Azure datacenter could occur, resulting in the combination of VM Sizes specified in the intent not being available in the data center. In such cases, an "OverconstrainedAllocationRequest" error will occur, even while deploying VMs of sizes specified in the intent. You can try deallocated all the resources in the proximity placement group and recreate them to get a data center that can accommodate the intent. If there is no datacenter with the specified VM Sizes after the decommissioning, you may have to modify the intent to use a different combination of VM Sizes, since the combination of VM sizes is no longer supported.
- Azure may retire an entire VM family or a specific set of VM sizes. If you have such a VM size in the intent, you may have to either remove it or replace it with a different size before the retirement date for the original VM size. Otherwise, the intent will no longer be valid.

## What to expect when using Proximity Placement Groups

Proximity placement groups offer colocation in the same data center. However, because proximity placement groups represent an additional deployment constraint, allocation failures can occur. There are few use cases where you may see allocation failures when using proximity placement groups:

- When you ask for the first virtual machine in the proximity placement group, the data center is automatically selected. In some cases, a second request for a different VM size, may fail if it doesn't exist in that data center. In this case, an **OverconstrainedAllocationRequest** error is returned. To avoid this error, try changing the order in which you deploy your VM sizes or have both resources deployed using a single ARM template.
- If the proximity placement group is created with intent, the VMs are not required to be deployed in any particular order and are not required to be batched using a single ARM template, since the intent is used to select a datacenter that supports all VM sizes indicated in the intent.
- In the case of elastic workloads, where you add and remove VM instances, having a proximity placement group constraint on your deployment may result in a failure to satisfy the request resulting in **AllocationFailure** error.
- Stopping (deallocate) and starting your VMs as needed is another way to achieve elasticity. Since the capacity is not kept once you stop (deallocate) a VM, starting it again may result in an **AllocationFailure** error.
- VM start and redeploy operations will continue to respect the Proximity Placement Group once successfully configured.

## Planned maintenance and Proximity Placement Groups

Planned maintenance events, like hardware decommissioning at an Azure datacenter, could potentially affect the alignment of resources in proximity placement groups. Resources may be moved to a different data center, disrupting the collocation and latency expectations associated with the proximity placement group.

### Check the alignment status

You can do the following to check the alignment status of your proximity placement groups.

- Proximity placement group colocation status can be viewed using the portal, CLI, and PowerShell.
  - PowerShell - colocation status can be obtained through `Get-AzProximityPlacementGroup` cmdlet by including the optional parameter '`-ColocationStatus`'.
  - CLI - colocation status can be obtained through `az ppg show` by including the optional parameter '`--include-colocation-status`'.
- For each proximity placement group, a **colocation status** property provides the current alignment

status summary of the grouped resources.

- **Aligned:** Resource is within the same latency envelop of the proximity placement group.
- **Unknown:** At least one of the VM resources are deallocated. After re-starting them successfully, the status should go back to **Aligned**.
- **Not aligned:** At least one VM resource is not aligned with the proximity placement group. The specific resources that are not aligned will also be called out separately in the membership section
- For Availability Sets, you can see information about alignment of individual VMs in the Availability Set Overview page.
- For scale sets, information about alignment of individual instances can be seen in the **Instances** tab of the Overview page for the scale set.

### Realign resources

If a proximity placement group is **Not Aligned**, you can stop\deallocate, and then restart the affected resources. If the VM is in an availability set or a scale set, all VMs in the availability set or scale set must be stopped\deallocated first before restarting them.

If there is an allocation failure due to deployment constraints, you may have to stop\deallocate all resources in the affected proximity placement group (including the aligned resources) first, and then restart them to restore alignment.

## Best practices

- For the lowest latency, use proximity placement groups together with accelerated networking. For more information, see [Create a Linux virtual machine with Accelerated Networking](#) or [Create a Windows virtual machine with Accelerated Networking](#).
- In order to avoid landing on hardware that doesn't support all the VM SKUs and sizes you require, use intent for proximity placement groups. If it is an already existing proximity placement group without intent, you can use a single ARM template with all VM sizes specified to avoid this issue.
- When reusing an existing placement group from which VMs were deleted, wait for the deletion to fully complete before adding VMs to it.
- If latency is your first priority, put VMs in a proximity placement group and the entire solution in an availability zone. But, if resiliency is your top priority, spread your instances across multiple availability zones (a single proximity placement group cannot span zones).

## Next steps

- Deploy a VM to a proximity placement group using the [Azure CLI](#) or [PowerShell](#).
- Learn how to [test network latency](#).
- Learn how to [optimize network throughput](#).
- Learn how to [use proximity placement groups with SAP applications](#).

# Deploy VMs to proximity placement groups using Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

To get VMs as close as possible, achieving the lowest possible latency, you should deploy them within a [proximity placement group](#).

A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

## Create the proximity placement group

Create a proximity placement group using `az ppg create`.

```
az group create --name myPPGGroup --location eastus
az ppg create \
  -n myPPG \
  -g myPPGGroup \
  -l eastus \
  -t standard
  --intent-vm-sizes Standard_E64s_v4 Standard_M416ms_v2 \
  -z 1
```

## List proximity placement groups

You can list all of your proximity placement groups using `az ppg list`.

```
az ppg list -o table
```

## Show proximity placement group

You can see the proximity placement group details and resources using `az ppg show`

```
az ppg show --name myPPG --resource-group myPPGGroup
{
  "availabilitySets": [],
  "colocationStatus": null,
  "id": "/subscriptions/[subscriptionId]/resourceGroups/myPPGGroup/providers/Microsoft.Compute/proximityPlacementGroups/MyPPG",
  "intent": {
    "vmSizes": [
      "Standard_E64s_v4",
      "Standard_M416ms_v2"
    ]
  },
  "location": "eastus",
  "name": "MyPPG",
  "proximityPlacementGroupType": "Standard",
  "resourceGroup": "myPPGGroup",
  "tags": {},
  "type": "Microsoft.Compute/proximityPlacementGroups",
  "virtualMachineScaleSets": [],
  "virtualMachines": [],
  "zones": [
    "1"
  ]
}
```

## Create a VM

Create a VM within the proximity placement group using [new az vm](#).

```
az vm create \
  -n myVM \
  -g myPPGGroup \
  --image UbuntuLTS \
  --ppg myPPG \
  --generate-ssh-keys \
  --size Standard_E64s_v4 \
  -l eastus
```

You can see the VM in the proximity placement group using [az ppg show](#).

```
az ppg show --name myppg --resource-group myppggroup --query "virtualMachines"
```

## Availability Sets

You can also create an availability set in your proximity placement group. Use the same `--ppg` parameter with [az vm availability-set create](#) to create an availability set and all of the VMs in the availability set will also be created in the same proximity placement group.

## Scale sets

You can also create a scale set in your proximity placement group. Use the same `--ppg` parameter with [az vmss create](#) to create a scale set and all of the instances will be created in the same proximity placement group.

## Next steps

Learn more about the [Azure CLI](#) commands for proximity placement groups.

# Deploy VMs to proximity placement groups using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

To get VMs as close as possible, achieving the lowest possible latency, you should deploy them within a [proximity placement group](#).

A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

## Create a proximity placement group

Create a proximity placement group using the [New-AzProximityPlacementGroup](#) cmdlet.

```
$resourceGroup = "myPPGResourceGroup"
$location = "East US"
$ppgName = "myPPG"
$zone = "1"
$vmSize1 = "Standard_E64s_v4"
$vmSize2 = "Standard_M416ms_v2"
New-AzResourceGroup -Name $resourceGroup -Location $location
$ppg = New-AzProximityPlacementGroup ` 
    -Location $location ` 
    -Name $ppgName ` 
    -ResourceGroupName $resourceGroup ` 
    -ProximityPlacementGroupType Standard 
    -Zone $zone 
    -IntentVmSizeList $vmSize1, $vmSize2
```

## List proximity placement groups

You can list all of the proximity placement groups using the [Get-AzProximityPlacementGroup](#) cmdlet.

```
Get-AzProximityPlacementGroup -ResourceGroupName $resourceGroup -Name $ppgName

ResourceGroupName      : myPPGResourceGroup
ProximityPlacementGroupType : Standard
Id                   :
/subscriptions/[subscriptionId]/resourceGroups/myPPGResourceGroup/providers/Microsoft.Compute/proximityPlace
mentGroups/myPPG
Name                 : myPPG
Type                : Microsoft.Compute/proximityPlacementGroups
Location             : eastus
Tags                : {}
Intent               :
VmSizes[0]           : Standard_E64s_v4
VmSizes[1]           : Standard_M416ms_v2
Zones[0]              : 1
```

## Create a VM

Create a VM in the proximity placement group using `-ProximityPlacementGroup $ppg.Id` to refer to the proximity placement group ID when you use [New-AzVM](#) to create the VM.

```
$vmName = "myVM"

New-AzVm `-
    -ResourceGroupName $resourceGroup `-
    -Name $vmName `-
    -Location $location `-
    -ProximityPlacementGroup $ppg.Id
```

You can see the VM in the placement group using [Get-AzProximityPlacementGroup](#).

```
Get-AzProximityPlacementGroup -ResourceId $ppg.Id | 
    Format-Table -Property VirtualMachines -Wrap
```

### Move an existing VM into a proximity placement group

You can also add an existing VM to a proximity placement group. You need to stop\deallocate the VM first, then update the VM and restart.

```
$ppg = Get-AzProximityPlacementGroup -ResourceGroupName myPPGResourceGroup -Name myPPG
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM
Stop-AzVM -Name $vm.Name -ResourceGroupName $vm.ResourceGroupName
Update-AzVM -VM $vm -ResourceGroupName $vm.ResourceGroupName -ProximityPlacementGroupId $ppg.Id
Start-AzVM -Name $vm.Name -ResourceGroupName $vm.ResourceGroupName
```

### Move an existing VM out of a proximity placement group

To remove a VM from a proximity placement group, you need to stop\deallocate the VM first, then update the VM and restart.

```
$ppg = Get-AzProximityPlacementGroup -ResourceGroupName myPPGResourceGroup -Name myPPG
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM
Stop-AzVM -Name $vm.Name -ResourceGroupName $vm.ResourceGroupName
$vm.ProximityPlacementGroup = ""
Update-AzVM -VM $vm -ResourceGroupName $vm.ResourceGroupName
Start-AzVM -Name $vm.Name -ResourceGroupName $vm.ResourceGroupName
```

## Availability Sets

You can also create an availability set in your proximity placement group. Use the same `-ProximityPlacementGroup` parameter with the [New-AzAvailabilitySet](#) cmdlet to create an availability set and all of the VMs created in the availability set will also be created in the same proximity placement group.

To add or remove an existing availability set to a proximity placement group, you first need to stop all of the VMs in the availability set.

### Move an existing availability set into a proximity placement group

```

$resourceGroup = "myResourceGroup"
$avSetName = "myAvailabilitySet"
$avSet = Get-AzAvailabilitySet -ResourceGroupName $resourceGroup -Name $avSetName
$vmIDs = $avSet.VirtualMachineReferences
foreach ($vmId in $vmIDs){
    $string = $vmID.Id.Split("/")
    $vmName = $string[8]
    Stop-AzVM -ResourceGroupName $resourceGroup -Name $vmName -Force
}

$ppg = Get-AzProximityPlacementGroup -ResourceGroupName myPPG -Name myPPG
Update-AzAvailabilitySet -AvailabilitySet $avSet -ProximityPlacementGroupId $ppg.Id
foreach ($vmId in $vmIDs){
    $string = $vmID.Id.Split("/")
    $vmName = $string[8]
    Start-AzVM -ResourceGroupName $resourceGroup -Name $vmName
}

```

## Move an existing availability set out of a proximity placement group

```

$resourceGroup = "myResourceGroup"
$avSetName = "myAvailabilitySet"
$avSet = Get-AzAvailabilitySet -ResourceGroupName $resourceGroup -Name $avSetName
$vmIDs = $avSet.VirtualMachineReferences
foreach ($vmId in $vmIDs){
    $string = $vmID.Id.Split("/")
    $vmName = $string[8]
    Stop-AzVM -ResourceGroupName $resourceGroup -Name $vmName -Force
}

$avSet.ProximityPlacementGroup = ""
Update-AzAvailabilitySet -AvailabilitySet $avSet
foreach ($vmId in $vmIDs){
    $string = $vmID.Id.Split("/")
    $vmName = $string[8]
    Start-AzVM -ResourceGroupName $resourceGroup -Name $vmName
}

```

## Scale sets

You can also create a scale set in your proximity placement group. Use the same `-ProximityPlacementGroup` parameter with [New-AzVmss](#) to create a scale set and all of the instances will be created in the same proximity placement group.

To add or remove an existing scale set to a proximity placement group, you first need to stop the scale set.

## Move an existing scale set into a proximity placement group

```

$ppg = Get-AzProximityPlacementGroup -ResourceGroupName myPPG -Name myPPG
$vmss = Get-AzVmss -ResourceGroupName myVMSResourceGroup -VMScaleSetName myScaleSet
Stop-AzVmss -VMScaleSetName $vmss.Name -ResourceGroupName $vmss.ResourceGroupName
Update-AzVmss -VMScaleSetName $vmss.Name -ResourceGroupName $vmss.ResourceGroupName -
ProximityPlacementGroupId $ppg.Id
Start-AzVmss -VMScaleSetName $vmss.Name -ResourceGroupName $vmss.ResourceGroupName

```

## Move an existing scale set out of a proximity placement group

```
$vmss = Get-AzVmss -ResourceGroupName myVMSSResourceGroup -VMScaleSetName myScaleSet
Stop-AzVmss -VMScaleSetName $vmss.Name -ResourceGroupName $vmss.ResourceGroupName
$vmss.ProximityPlacementGroup = ""
Update-AzVmss -VirtualMachineScaleSet $vmss -VMScaleSetName $vmss.Name -ResourceGroupName
$vmss.ResourceGroupName
Start-AzVmss -VMScaleSetName $vmss.Name -ResourceGroupName $vmss.ResourceGroupName
```

## Next steps

You can also use the [Azure CLI](#) to create proximity placement groups.

# Create a proximity placement group using the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

To get VMs as close as possible, achieving the lowest possible latency, you should deploy them within a [proximity placement group](#).

A proximity placement group is a logical grouping used to make sure that Azure compute resources are physically located close to each other. Proximity placement groups are useful for workloads where low latency is a requirement.

## NOTE

Proximity placement groups cannot be used with dedicated hosts.

Intent for proximity placement groups is not supported on Azure portal. Use ARM templates or other client tools like Powershell or CLI to provide intent for proximity placement groups.

If you want to use availability zones together with placement groups, you need to make sure that the VMs in the placement group are also all in the same availability zone.

## Create the proximity placement group

1. Type **proximity placement group** in the search.
2. Under **Services** in the search results, select **Proximity placement groups**.
3. In the **Proximity placement groups** page, select **Add**.
4. In the **Basics** tab, under **Project details**, make sure the correct subscription is selected.
5. In **Resource group** either select **Create new** to create a new group or select an empty resource group that already exists, from the drop-down.
6. In **Region** select the location where you want the proximity placement group to be created.
7. In **Proximity placement group name** type a name and then select **Review + create**.
8. After validation passes, select **Create** to create the proximity placement group.

Home > New > Proximity Placement Group > Create Proximity Placement Group

## Create Proximity Placement Group

[Basics](#) [Tags](#) [Review + create](#)

Fill out the required fields and then review the information on the Review + create tab. Once you're satisfied, click Create to deploy.

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Pay-As-You-Go
Resource group *	(New) myPPG
	<a href="#">Create new</a>

**Instance details**

Region *	East US
Proximity placement group name *	myPPG

[Review + create](#) [Previous](#) [Next : Tags >](#)

## Create a VM

1. While creating a VM in the portal, go to the **Advanced** tab.
2. In the **Proximity placement group** selection, select the correct placement group.

**Proximity placement group**

Proximity placement groups allow you to group Azure resources physically closer together in the same region. [Learn more](#)

Proximity placement group	myPPG
---------------------------	-------

3. When you are done making all of the other required selections, select **Review + create**.
4. After it passes validation, select **Create** to deploy the VM in the placement group.

## Add VMs in an availability set to a proximity placement group

If the VM is part of the Availability set, you need to add the availability set into the the placement group, before adding the VMs.

1. In the [portal](#) search for *Availability sets* and select your availability set from the results.
2. Stop\deallocate each VM in the availability set by selecting the VM, then selecting **Stop** on the page for the VM, and then select **OK** to stop the VM.
3. On the page for your availability set, make sure all of the VMs have the **Status** listed as **Stopped (deallocated)**.
4. In the left menu, select **Configuration**.
5. Under **Proximity placement group**, select a placement group from the drop-down, and then select **Save**.
6. Select **Overview** from the left menu to see the list of VMs again.
7. Select each VM in the availability set, and then select **Start** on the page for each VM.

## Add existing VM to placement group

1. On the page for the VM, select **Stop**.
2. Once the status of the VM is listed as **Stopped (deallocated)**, select **Configuration** on the left menu.
3. Under **Proximity placement group**, select a placement group from the drop-down, and then select **Save**.
4. Select **Overview** from the left menu, then select **Start** to restart the VM.

## Next steps

You can also use the [Azure PowerShell](#) to create proximity placement groups.

# Understand VM reboots - maintenance vs. downtime

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

There are three scenarios that can lead to virtual machines in Azure being impacted: unplanned hardware maintenance, unexpected downtime, and planned maintenance.

## Unplanned hardware maintenance event

Unplanned hardware maintenance occurs when the Azure platform predicts that the hardware or any platform component associated to a physical machine, is about to fail. When the platform predicts a failure, it will issue an unplanned hardware maintenance event to reduce the impact to the virtual machines hosted on that hardware. Azure uses [Live Migration](#) technology to migrate the Virtual Machines from the failing hardware to a healthy physical machine. Live Migration is a VM preserving operation that only pauses the Virtual Machine for a short time. Memory, open files, and network connections are maintained, but performance might be reduced before and/or after the event. In cases where Live Migration cannot be used, the VM will experience Unexpected Downtime, as described below.

## Unexpected downtime

Unexpected downtime is when the hardware or the physical infrastructure for the virtual machine fails unexpectedly. This can include local network failures, local disk failures, or other rack level failures. When detected, the Azure platform automatically migrates (heals) your virtual machine to a healthy physical machine in the same data center. During the healing procedure, virtual machines experience downtime (reboot) and in some cases loss of the temporary drive. The attached OS and data disks are always preserved.

Virtual machines can also experience downtime in the unlikely event of an outage or disaster that affects an entire data center, or even an entire region. For these scenarios, Azure provides protection options including [availability zones](#) and [paired regions](#).

## Planned maintenance events

Planned maintenance events are periodic updates made by Microsoft to the underlying Azure platform to improve overall reliability, performance, and security of the platform infrastructure that your virtual machines run on. Most of these updates are performed without any impact upon your Virtual Machines or Cloud Services (see [Maintenance that doesn't require a reboot](#)). While the Azure platform attempts to use VM Preserving Maintenance in all possible occasions, there are rare instances when these updates require a reboot of your virtual machine to apply the required updates to the underlying infrastructure. In this case, you can perform Azure Planned Maintenance with Maintenance-Redeploy operation by initiating the maintenance for their VMs in the suitable time window. For more information, see [Planned Maintenance for Virtual Machines](#).

## Reduce downtime

To reduce the impact of downtime due to one or more of these events, we recommend the following high availability best practices for your virtual machines:

- Use [Availability Zones](#) to protect from data center failures
- Configure multiple virtual machines in an [availability set](#) for redundancy

- Use [scheduled events for Linux](#) or [scheduled events for Windows](#) to proactively respond to VM impacting events
- Configure each application tier into separate availability sets
- Combine a [load balancer](#) with availability zones or sets

## Next steps

To learn more about availability options in Azure see, see [Availability overview](#).

# Regions for virtual machines in Azure

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

It is important to understand how and where your virtual machines (VMs) operate in Azure, along with your options to maximize performance, availability, and redundancy. This article provides you with an overview of the availability and redundancy features of Azure.

## What are Azure regions?

Azure operates in multiple datacenters around the world. These datacenters are grouped in to geographic regions, giving you flexibility in choosing where to build your applications.

You create Azure resources in defined geographic regions like 'West US', 'North Europe', or 'Southeast Asia'. You can review the [list of regions and their locations](#). Within each region, multiple datacenters exist to provide for redundancy and availability. This approach gives you flexibility as you design applications to create VMs closest to your users and to meet any legal, compliance, or tax purposes.

## Special Azure regions

Azure has some special regions that you may wish to use when building out your applications for compliance or legal purposes. These special regions include:

- **US Gov Virginia and US Gov Iowa**
  - A physical and logical network-isolated instance of Azure for US government agencies and partners, operated by screened US persons. Includes additional compliance certifications such as [FedRAMP](#) and [DISA](#). Read more about [Azure Government](#).
- **China East and China North**
  - These regions are available through a unique partnership between Microsoft and 21Vianet, whereby Microsoft does not directly maintain the datacenters. See more about [Azure China 21Vianet](#).
- **Germany Central and Germany Northeast**
  - These regions are available via a data trustee model whereby customer data remains in Germany under control of T-Systems, a Deutsche Telekom company, acting as the German data trustee.

## Region pairs

Each Azure region is paired with another region within the same geography (such as US, Europe, or Asia). This approach allows for the replication of resources, such as VM storage, across a geography that should reduce the likelihood of natural disasters, civil unrest, power outages, or physical network outages affecting both regions at once. Additional advantages of region pairs include:

- In the event of a wider Azure outage, one region is prioritized out of every pair to help reduce the time to restore for applications.
- Planned Azure updates are rolled out to paired regions one at a time to minimize downtime and risk of application outage.
- Data continues to reside within the same geography as its pair (except for Brazil South) for tax and law enforcement jurisdiction purposes.

Examples of region pairs include:

PRIMARY	SECONDARY
West US	East US
North Europe	West Europe
Southeast Asia	East Asia

You can see the full [list of regional pairs here](#).

## Feature availability

Some services or VM features are only available in certain regions, such as specific VM sizes or storage types. There are also some global Azure services that do not require you to select a particular region, such as [Azure Active Directory](#), [Traffic Manager](#), or [Azure DNS](#). To assist you in designing your application environment, you can check the [availability of Azure services across each region](#). You can also [programmatically query the supported VM sizes and restrictions in each region](#).

## Storage availability

Understanding Azure regions and geographies becomes important when you consider the available storage replication options. Depending on the storage type, you have different replication options.

### Azure Managed Disks

- Locally redundant storage (LRS)
  - Replicates your data three times within the region in which you created your storage account.

### Storage account-based disks

- Locally redundant storage (LRS)
  - Replicates your data three times within the region in which you created your storage account.
- Zone redundant storage (ZRS)
  - Replicates your data three times across two to three facilities, either within a single region or across two regions.
- Geo-redundant storage (GRS)
  - Replicates your data to a secondary region that is hundreds of miles away from the primary region.
- Read-access geo-redundant storage (RA-GRS)
  - Replicates your data to a secondary region, as with GRS, but also then provides read-only access to the data in the secondary location.

The following table provides a quick overview of the differences between the storage replication types:

REPLICATION STRATEGY	LRS	ZRS	GRS	RA-GRS
Data is replicated across multiple facilities.	No	Yes	Yes	Yes
Data can be read from the secondary location and from the primary location.	No	No	No	Yes

REPLICATION STRATEGY	LRS	ZRS	GRS	RA-GRS
Number of copies of data maintained on separate nodes.	3	3	6	6

You can read more about [Azure Storage replication options here](#). For more information about managed disks, see [Azure Managed Disks overview](#).

### Storage costs

Prices vary depending on the storage type and availability that you select.

#### Azure Managed Disks

- Premium Managed Disks are backed by Solid-State Drives (SSDs) and Standard Managed Disks are backed by regular spinning disks. Both Premium and Standard Managed Disks are charged based on the provisioned capacity for the disk.

#### Unmanaged disks

- Premium storage is backed by Solid-State Drives (SSDs) and is charged based on the capacity of the disk.
- Standard storage is backed by regular spinning disks and is charged based on the in-use capacity and desired storage availability.
  - For RA-GRS, there is an additional Geo-Replication Data Transfer charge for the bandwidth of replicating that data to another Azure region.

See [Azure Storage Pricing](#) for pricing information on the different storage types and availability options.

## Next steps

For more information, see [Azure regions](#).

# Availability sets overview

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

## NOTE

We recommend that new customers choose [virtual machine scale sets with flexible orchestration mode](#) for high availability with the widest range of features. Virtual machine scale sets allow VM instances to be centrally managed, configured, and updated, and will automatically increase or decrease the number of VM instances in response to demand or a defined schedule. Availability sets only offer high availability.

This article provides you with an overview of the availability features of Azure virtual machines (VMs).

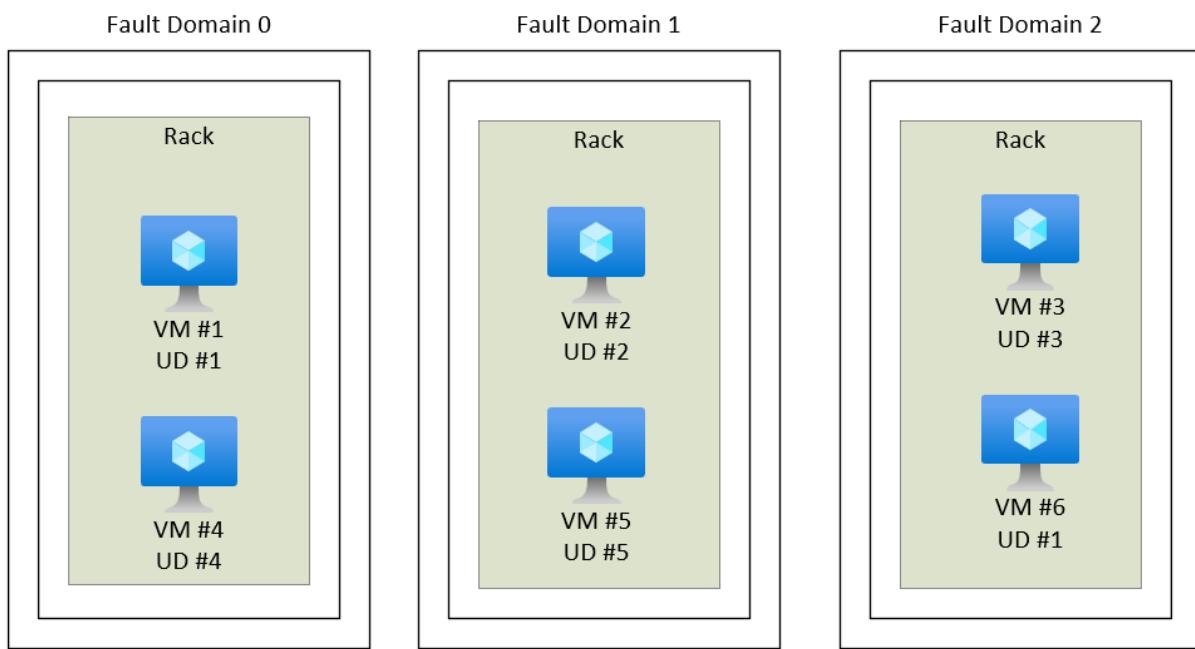
## What is an availability set?

An availability set is a logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability. We recommended that two or more VMs are created within an availability set to provide for a highly available application and to meet the [99.95% Azure SLA](#). There is no cost for the Availability Set itself, you only pay for each VM instance that you create.

## How do availability sets work?

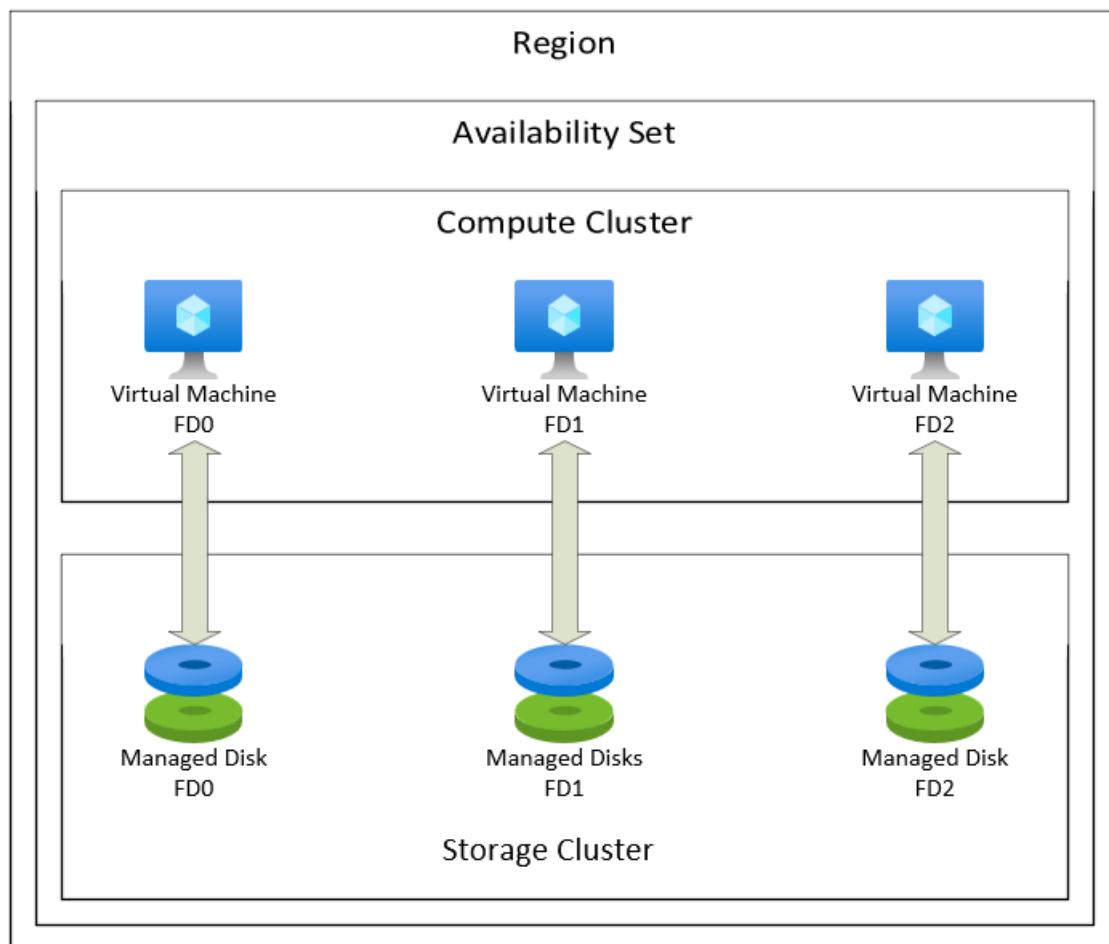
Each virtual machine in your availability set is assigned an **update domain** and a **fault domain** by the underlying Azure platform. Each availability set can be configured with up to three fault domains and twenty update domains. These configurations cannot be changed once the availability set has been created. Update domains indicate groups of virtual machines and underlying physical hardware that can be rebooted at the same time. When more than five virtual machines are configured within a single availability set with five update domains, the sixth virtual machine is placed into the same update domain as the first virtual machine, the seventh in the same update domain as the second virtual machine, and so on. The order of update domains being rebooted may not proceed sequentially during planned maintenance, but only one update domain is rebooted at a time. A rebooted update domain is given 30 minutes to recover before maintenance is initiated on a different update domain.

Fault domains define the group of virtual machines that share a common power source and network switch. By default, the virtual machines configured within your availability set are separated across up to three fault domains. While placing your virtual machines into an availability set does not protect your application from operating system or application-specific failures, it does limit the impact of potential physical hardware failures, network outages, or power interruptions.



VMs are also aligned with disk fault domains. This alignment ensures that all the managed disks attached to a VM are within the same fault domains.

Only VMs with managed disks can be created in a managed availability set. The number of managed disk fault domains varies by region - either two or three managed disk fault domains per region.



## Next steps

You can now start to use these availability and redundancy features to build your Azure environment. For best

practices information, see [Azure availability best practices](#).

# Create and deploy virtual machines in an availability set using Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

In this tutorial, you learn how to increase the availability and reliability of your Virtual Machine solutions on Azure using a capability called Availability Sets. Availability sets ensure that the VMs you deploy on Azure are distributed across multiple isolated hardware clusters. Doing this ensures that if a hardware or software failure within Azure happens, only a subset of your VMs is impacted and that your overall solution remains available and operational.

In this tutorial, you learn how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes

This tutorial uses the CLI within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

If you choose to install and use the CLI locally, this tutorial requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create an availability set

You can create an availability set using `az vm availability-set create`. In this example, the number of update and fault domains is set to 2 for the availability set named *myAvailabilitySet* in the *myResourceGroupAvailability* resource group.

First, create a resource group with [az group create](#), then create the availability set:

```
az group create --name myResourceGroupAvailability --location eastus

az vm availability-set create \
    --resource-group myResourceGroupAvailability \
    --name myAvailabilitySet \
    --platform-fault-domain-count 2 \
    --platform-update-domain-count 2
```

Availability Sets allow you to isolate resources across fault domains and update domains. A **fault domain** represents an isolated collection of server + network + storage resources. In the preceding example, the availability set is distributed across at least two fault domains when the VMs are deployed. The availability set is also distributed across two **update domains**. Two update domains ensure that when Azure performs software updates, the VM resources are isolated, preventing all the software that runs on the VM from being updated at the same time.

## Create VMs inside an availability set

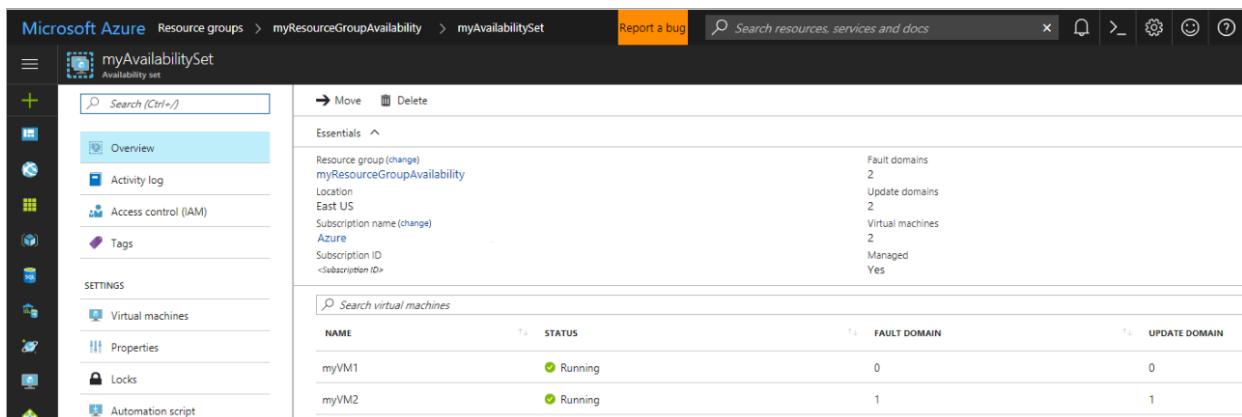
VMs must be created within the availability set to make sure they are correctly distributed across the hardware. An existing VM cannot be added to an availability set after it is created.

When a VM is created with [az vm create](#), use the `--availability-set` parameter to specify the name of the availability set.

```
for i in `seq 1 2`; do
    az vm create \
        --resource-group myResourceGroupAvailability \
        --name myVM$i \
        --availability-set myAvailabilitySet \
        --size Standard_DS1_v2 \
        --vnet-name myVnet \
        --subnet mySubnet \
        --image UbuntuLTS \
        --admin-username azureuser \
        --generate-ssh-keys
done
```

There are now two virtual machines within the availability set. Because they are in the same availability set, Azure ensures that the VMs and all their resources (including data disks) are distributed across isolated physical hardware. This distribution helps ensure much higher availability of the overall VM solution.

The availability set distribution can be viewed in the portal by going to Resource Groups > myResourceGroupAvailability > myAvailabilitySet. The VMs are distributed across the two fault and update domains, as shown in the following example:



NAME	STATUS	FAULT DOMAIN	UPDATE DOMAIN
myVM1	Running	0	0
myVM2	Running	1	1

## Check for available VM sizes

Additional VMs can be added to the availability set later, where VM sizes are available on the hardware. Use [az vm availability-set list-sizes](#) to list all the available sizes on the hardware cluster for the availability set:

```
az vm availability-set list-sizes \
    --resource-group myResourceGroupAvailability \
    --name myAvailabilitySet \
    --output table
```

## Next steps

In this tutorial, you learned how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes

Advance to the next tutorial to learn about virtual machine scale sets.

[Create a virtual machine scale set](#)

- To learn more about availability zones, visit the [Availability Zones documentation](#).
- More documentation about both availability sets and availability zones is also available at [Availability options for Azure Virtual Machines](#).
- To try out availability zones, visit [Create a Linux virtual machine in an availability zone with the Azure CLI](#)

# Create and deploy virtual machines in an availability set using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this tutorial, you learn how to increase the availability and reliability of your Virtual Machines (VMs) using Availability Sets. Availability Sets make sure the VMs you deploy on Azure are distributed across multiple, isolated hardware nodes, in a cluster.

In this tutorial, you learn how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes
- Check Azure Advisor

## Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

## Create an availability set

The hardware in a location is divided in to multiple update domains and fault domains. An **update domain** is a group of VMs and underlying physical hardware that can be rebooted at the same time. VMs in the same **fault domain** share common storage as well as a common power source and network switch.

You can create an availability set using [New-AzAvailabilitySet](#). In this example, the number of both update and fault domains is 2 and the availability set is named *myAvailabilitySet*.

Create a resource group.

```
New-AzResourceGroup ` 
-Name myResourceGroupAvailability ` 
-Location EastUS
```

Create a managed availability set using [New-AzAvailabilitySet](#) with the `-sku aligned` parameter.

```
New-AzAvailabilitySet ` 
-Location "EastUS" ` 
-Name "myAvailabilitySet" ` 
-ResourceGroupName "myResourceGroupAvailability" ` 
-Sku aligned ` 
-PlatformFaultDomainCount 2 ` 
-PlatformUpdateDomainCount 2
```

# Create VMs inside an availability set

VMs must be created within the availability set to make sure they're correctly distributed across the hardware. You can't add an existing VM to an availability set after it's created.

When you create a VM with [New-AzVM](#), you use the `-AvailabilitySetName` parameter to specify the name of the availability set.

First, set an administrator username and password for the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Now create two VMs with [New-AzVM](#) in the availability set.

```
for ($i=1; $i -le 2; $i++)
{
    New-AzVm ` 
        -ResourceGroupName "myResourceGroupAvailability" ` 
        -Name "myVM$i" ` 
        -Location "East US" ` 
        -VirtualNetworkName "myVnet" ` 
        -SubnetName "mySubnet" ` 
        -SecurityGroupName "myNetworkSecurityGroup" ` 
        -PublicIpAddressName "myPublicIpAddress$i" ` 
        -AvailabilitySetName "myAvailabilitySet" ` 
        -Credential $cred
}
```

It takes a few minutes to create and configure both VMs. When finished, you have two virtual machines distributed across the underlying hardware.

If you look at the availability set in the portal by going to **Resource Groups > myResourceGroupAvailability > myAvailabilitySet**, you should see how the VMs are distributed across the two fault and update domains.

The screenshot shows the Azure portal interface for managing an availability set. The top navigation bar includes 'Microsoft Azure', 'Resource groups', 'myResourceGroupAvailability', 'myAvailabilitySet', and a search bar. The main content area displays the 'myAvailabilitySet' details under the 'Essentials' tab. It lists the resource group as 'myResourceGroupAvailability', location as 'East US', subscription name as 'Azure', and subscription ID as '<Subscription ID>'. It also indicates 'Fault domains: 2', 'Update domains: 2', 'Virtual machines: 2', and 'Managed: Yes'. Below this, a table titled 'Search virtual machines' lists two entries: 'myVM1' and 'myVM2', both marked as 'Running'. The table includes columns for NAME, STATUS, FAULT DOMAIN, and UPDATE DOMAIN.

NAME	STATUS	FAULT DOMAIN	UPDATE DOMAIN
myVM1	Running	0	0
myVM2	Running	1	1

#### NOTE

Under certain circumstances, 2 VMs in the same AvailabilitySet could share the same FaultDomain. This can be confirmed by going into your availability set and checking the Fault Domain column. This can be caused by the following sequence of events while deploying the VMs:

1. The 1st VM is Deployed
2. The 1st VM is Stopped/Deallocated
3. The 2nd VM is Deployed. Under these circumstances, the OS Disk of the 2nd VM might be created on the same Fault Domain as the 1st VM, and so the 2nd VM will also land on the same FaultDomain. To avoid this issue, it's recommended to not stop/deallocate the VMs between deployments.

## Check for available VM sizes

When you create a VM inside a availability set, you need to know what VM sizes are available on the hardware. Use [Get-AzVMSize](#) command to get all available sizes for virtual machines that you can deploy in the availability set.

```
Get-AzVMSize ` 
    -ResourceGroupName "myResourceGroupAvailability" ` 
    -AvailabilitySetName "myAvailabilitySet"
```

## Check Azure Advisor

You can also use Azure Advisor to get more information on how to improve the availability of your VMs. Azure Advisor analyzes your configuration and usage telemetry, then recommends solutions that can help you improve the cost effectiveness, performance, availability, and security of your Azure resources.

Sign in to the [Azure portal](#), select **All services**, and type **Advisor**. The Advisor dashboard shows personalized recommendations for the selected subscription. For more information, see [Get started with Azure Advisor](#).

## Next steps

In this tutorial, you learned how to:

- Create an availability set
- Create a VM in an availability set
- Check available VM sizes
- Check Azure Advisor

Advance to the next tutorial to learn about virtual machine scale sets.

[Create a VM scale set](#)

# Change the availability set for a VM using Azure PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

The following steps describe how to change the availability set of a VM using Azure PowerShell. A VM can only be added to an availability set when it is created. To change the availability set, you need to delete and then recreate the virtual machine.

This article was last tested on 2/12/2019 using the [Azure Cloud Shell](#) and the [Az PowerShell module](#) version 1.2.0.

## WARNING

This is just an example and in some cases it will need to be updated for your specific deployment.

Make sure the disks are set to `detach` as the `delete` option. If they are set to `delete`, update the VMs before deleting the VMs.

If your VM is attached to a load balancer, you will need to update the script to handle that case.

Some extensions may also need to be reinstalled after you finish this process.

If your VM uses hybrid benefits, you will need to update the example to enable hybrid benefits on the new VM.

## Change the availability set

The following script provides an example of gathering the required information, deleting the original VM and then recreating it in a new availability set.

```
# Set variables
$resourceGroup = "myResourceGroup"
$vmName = "myVM"
$newAvailSetName = "myAvailabilitySet"

# Get the details of the VM to be moved to the Availability Set
$originalVM = Get-AzVM `-
    -ResourceGroupName $resourceGroup `-
    -Name $vmName

# Create new availability set if it does not exist
$availSet = Get-AzAvailabilitySet `-
    -ResourceGroupName $resourceGroup `-
    -Name $newAvailSetName `-
    -ErrorAction Ignore
if (-Not $availSet) {
    $availSet = New-AzAvailabilitySet `-
        -Location $originalVM.Location `-
        -Name $newAvailSetName `-
        -ResourceGroupName $resourceGroup `-
        -PlatformFaultDomainCount 2 `-
        -PlatformUpdateDomainCount 2 `-
        -Sku Aligned
}

# Remove the original VM
```

```

Remove-AzVM -ResourceGroupName $resourceGroup -Name $vmName

# Create the basic configuration for the replacement VM.
$newVM = New-AzVMConfig ` 
    -VMName $originalVM.Name ` 
    -VmSize $originalVM.HardwareProfile.VmSize ` 
    -AvailabilitySetId $availSet.Id

# For a Linux VM, change the last parameter from -Windows to -Linux
Set-AzVMOSDisk ` 
    -VM $newVM -CreateOption Attach ` 
    -ManagedDiskId $originalVM.StorageProfile.OsDisk.ManagedDisk.Id ` 
    -Name $originalVM.StorageProfile.OsDisk.Name ` 
    -Windows

# Add Data Disks
foreach ($disk in $originalVM.StorageProfile.DataDisks) {
    Add-AzVMDisk -VM $newVM ` 
        -Name $disk.Name ` 
        -ManagedDiskId $disk.ManagedDisk.Id ` 
        -Caching $disk.Caching ` 
        -Lun $disk.Lun ` 
        -DiskSizeInGB $disk.DiskSizeGB ` 
        -CreateOption Attach
}

# Add NIC(s) and keep the same NIC as primary; keep the Private IP too, if it exists.
foreach ($nic in $originalVM.NetworkProfile.NetworkInterfaces) {
    if ($nic.Primary -eq "True")
    {
        Add-AzVMNetworkInterface ` 
            -VM $newVM ` 
            -Id $nic.Id -Primary
    }
    else
    {
        Add-AzVMNetworkInterface ` 
            -VM $newVM ` 
            -Id $nic.Id
    }
}

# Recreate the VM
New-AzVM ` 
    -ResourceGroupName $resourceGroup ` 
    -Location $originalVM.Location ` 
    -VM $newVM ` 
    -DisableBginfoExtension

```

## Next steps

Add additional storage to your VM by adding an additional [data disk](#).

# Introduction to Azure managed disks

9/21/2022 • 12 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure managed disks are block-level storage volumes that are managed by Azure and used with Azure Virtual Machines. Managed disks are like a physical disk in an on-premises server but, virtualized. With managed disks, all you have to do is specify the disk size, the disk type, and provision the disk. Once you provision the disk, Azure handles the rest.

The available types of disks are ultra disks, premium solid-state drives (SSD), standard SSDs, and standard hard disk drives (HDD). For information about each individual disk type, see [Select a disk type for IaaS VMs](#).

## Benefits of managed disks

Let's go over some of the benefits you gain by using managed disks.

### Highly durable and available

Managed disks are designed for 99.999% availability. Managed disks achieve this by providing you with three replicas of your data, allowing for high durability. If one or even two replicas experience issues, the remaining replicas help ensure persistence of your data and high tolerance against failures. This architecture has helped Azure consistently deliver enterprise-grade durability for infrastructure as a service (IaaS) disks, with an industry-leading ZERO% annualized failure rate.

### Simple and scalable VM deployment

Using managed disks, you can create up to 50,000 VM **disks** of a type in a subscription per region, allowing you to create thousands of **VMs** in a single subscription. This feature also further increases the scalability of [virtual machine scale sets](#) by allowing you to create up to 1,000 VMs in a virtual machine scale set using a Marketplace image.

### Integration with availability sets

Managed disks are integrated with availability sets to ensure that the disks of [VMs in an availability set](#) are sufficiently isolated from each other to avoid a single point of failure. Disks are automatically placed in different storage scale units (stamps). If a stamp fails due to hardware or software failure, only the VM instances with disks on those stamps fail. For example, let's say you have an application running on five VMs, and the VMs are in an Availability Set. The disks for those VMs won't all be stored in the same stamp, so if one stamp goes down, the other instances of the application continue to run.

### Integration with Availability Zones

Managed disks support [Availability Zones](#), which is a high-availability offering that protects your applications from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. With Availability Zones, Azure offers industry best 99.99% VM uptime SLA.

### Azure Backup support

To protect against regional disasters, [Azure Backup](#) can be used to create a backup job with time-based backups and backup retention policies. This allows you to perform VM or managed disk restorations at will. Currently Azure Backup supports disk sizes up to 32 tebibyte (TiB) disks. [Learn more](#) about Azure VM backup support.

### Azure Disk Backup

Azure Backup offers Azure Disk Backup (preview) as a native, cloud-based backup solution that protects your data in managed disks. It's a simple, secure, and cost-effective solution that enables you to configure protection for managed disks in a few steps. Azure Disk Backup offers a turnkey solution that provides snapshot lifecycle management for managed disks by automating periodic creation of snapshots and retaining it for configured duration using backup policy. For details on Azure Disk Backup, see [Overview of Azure Disk Backup \(in preview\)](#).

## Granular access control

You can use [Azure role-based access control \(Azure RBAC\)](#) to assign specific permissions for a managed disk to one or more users. Managed disks expose a variety of operations, including read, write (create/update), delete, and retrieving a [shared access signature \(SAS\) URI](#) for the disk. You can grant access to only the operations a person needs to perform their job. For example, if you don't want a person to copy a managed disk to a storage account, you can choose not to grant access to the export action for that managed disk. Similarly, if you don't want a person to use an SAS URI to copy a managed disk, you can choose not to grant that permission to the managed disk.

## Upload your vhd

Direct upload makes it easy to transfer your vhd to an Azure managed disk. Previously, you had to follow a more involved process that included staging your data in a storage account. Now, there are fewer steps. It is easier to upload on premises VMs to Azure, upload to large managed disks, and the backup and restore process is simplified. It also reduces cost by allowing you to upload data to managed disks directly without attaching them to VMs. You can use direct upload to upload vhds up to 32 TiB in size.

To learn how to transfer your vhd to Azure, see the [CLI](#) or [PowerShell](#) articles.

# Security

## Private Links

Private Link support for managed disks can be used to import or export a managed disk internal to your network. Private Links allow you to generate a time bound Shared Access Signature (SAS) URI for unattached managed disks and snapshots that you can use to export the data to other regions for regional expansion, disaster recovery, and forensic analysis. You can also use the SAS URI to directly upload a VHD to an empty disk from on-premises. Now you can leverage [Private Links](#) to restrict the export and import of managed disks so that it can only occur within your Azure virtual network. Private Links allows you to ensure your data only travels within the secure Microsoft backbone network.

To learn how to enable Private Links for importing or exporting a managed disk, see the [CLI](#) or [Portal](#) articles.

## Encryption

Managed disks offer two different kinds of encryption. The first is Server Side Encryption (SSE), which is performed by the storage service. The second one is Azure Disk Encryption (ADE), which you can enable on the OS and data disks for your VMs.

### Server-side encryption

Server-side encryption provides encryption-at-rest and safeguards your data to meet your organizational security and compliance commitments. Server-side encryption is enabled by default for all managed disks, snapshots, and images, in all the regions where managed disks are available. (Temporary disks, on the other hand, are not encrypted by server-side encryption unless you enable encryption at host; see [Disk Roles: temporary disks](#)).

You can either allow Azure to manage your keys for you, these are platform-managed keys, or you can manage the keys yourself, these are customer-managed keys. Visit the [Server-side encryption of Azure Disk Storage](#) article for details.

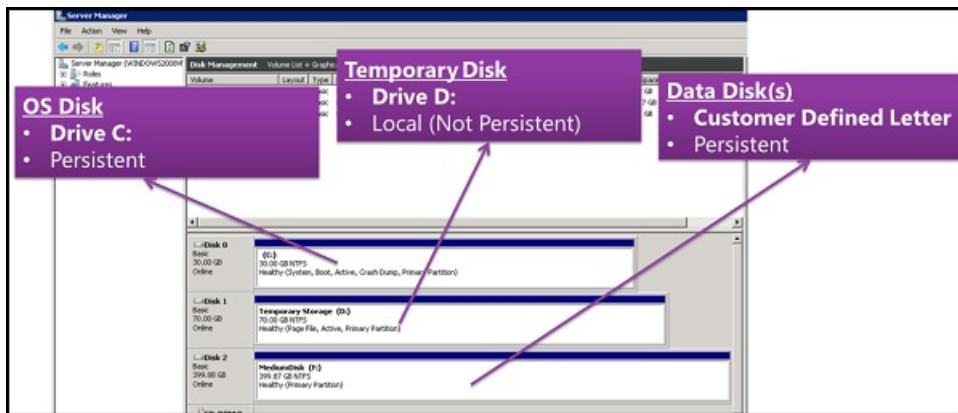
### Azure Disk Encryption

Azure Disk Encryption allows you to encrypt the OS and Data disks used by an IaaS Virtual Machine. This

encryption includes managed disks. For Windows, the drives are encrypted using industry-standard BitLocker encryption technology. For Linux, the disks are encrypted using the DM-Crypt technology. The encryption process is integrated with Azure Key Vault to allow you to control and manage the disk encryption keys. For more information, see [Azure Disk Encryption for Linux VMs](#) or [Azure Disk Encryption for Windows VMs](#).

## Disk roles

There are three main disk roles in Azure: the data disk, the OS disk, and the temporary disk. These roles map to disks that are attached to your virtual machine.



### Data disk

A data disk is a managed disk that's attached to a virtual machine to store application data, or other data you need to keep. Data disks are registered as SCSI drives and are labeled with a letter that you choose. Each data disk has a maximum capacity of 32,767 gibibytes (GiB). The size of the virtual machine determines how many data disks you can attach to it and the type of storage you can use to host the disks.

### OS disk

Every virtual machine has one attached operating system disk. That OS disk has a pre-installed OS, which was selected when the VM was created. This disk contains the boot volume.

This disk has a maximum capacity of 4,095 GiB, however, many operating systems are partitioned with [master boot record \(MBR\)](#) by default. MBR limits the usable size to 2 TiB. If you need more than 2 TiB, create and attach [data disks](#) and use them for data storage. If you need to store data on the OS disk and require the additional space, [convert it to GUID Partition Table \(GPT\)](#). To learn about the differences between MBR and GPT on Windows deployments, see [Windows and GPT FAQ](#).

### Temporary disk

Most VMs contain a temporary disk, which is not a managed disk. The temporary disk provides short-term storage for applications and processes, and is intended to only store data such as page or swap files. Data on the temporary disk may be lost during a [maintenance event](#) or when you [redeploy a VM](#). During a successful standard reboot of the VM, data on the temporary disk will persist. For more information about VMs without temporary disks, see [Azure VM sizes with no local temporary disk](#).

On Azure Linux VMs, the temporary disk is typically /dev/sdb and on Windows VMs the temporary disk is D: by default. The temporary disk is not encrypted by server side encryption unless you enable encryption at host.

## Managed disk snapshots

A managed disk snapshot is a read-only crash-consistent full copy of a managed disk that is stored as a standard managed disk by default. With snapshots, you can back up your managed disks at any point in time. These snapshots exist independent of the source disk and can be used to create new managed disks.

Snapshots are billed based on the used size. For example, if you create a snapshot of a managed disk with

provisioned capacity of 64 GiB and actual used data size of 10 GiB, that snapshot is billed only for the used data size of 10 GiB. You can see the used size of your snapshots by looking at the [Azure usage report](#). For example, if the used data size of a snapshot is 10 GiB, the **daily** usage report will show  $10 \text{ GiB}/(31 \text{ days}) = 0.3226$  as the consumed quantity.

To learn more about how to create snapshots for managed disks, see the [Create a snapshot of a managed disk](#) article.

## Images

Managed disks also support creating a managed custom image. You can create an image from your custom VHD in a storage account or directly from a generalized (sysprepped) VM. This process captures a single image. This image contains all managed disks associated with a VM, including both the OS and data disks. This managed custom image enables creating hundreds of VMs using your custom image without the need to copy or manage any storage accounts.

For information on creating images, see the following articles:

- [How to capture a managed image of a generalized VM in Azure](#)
- [How to generalize and capture a Linux virtual machine using the Azure CLI](#)

### Images versus snapshots

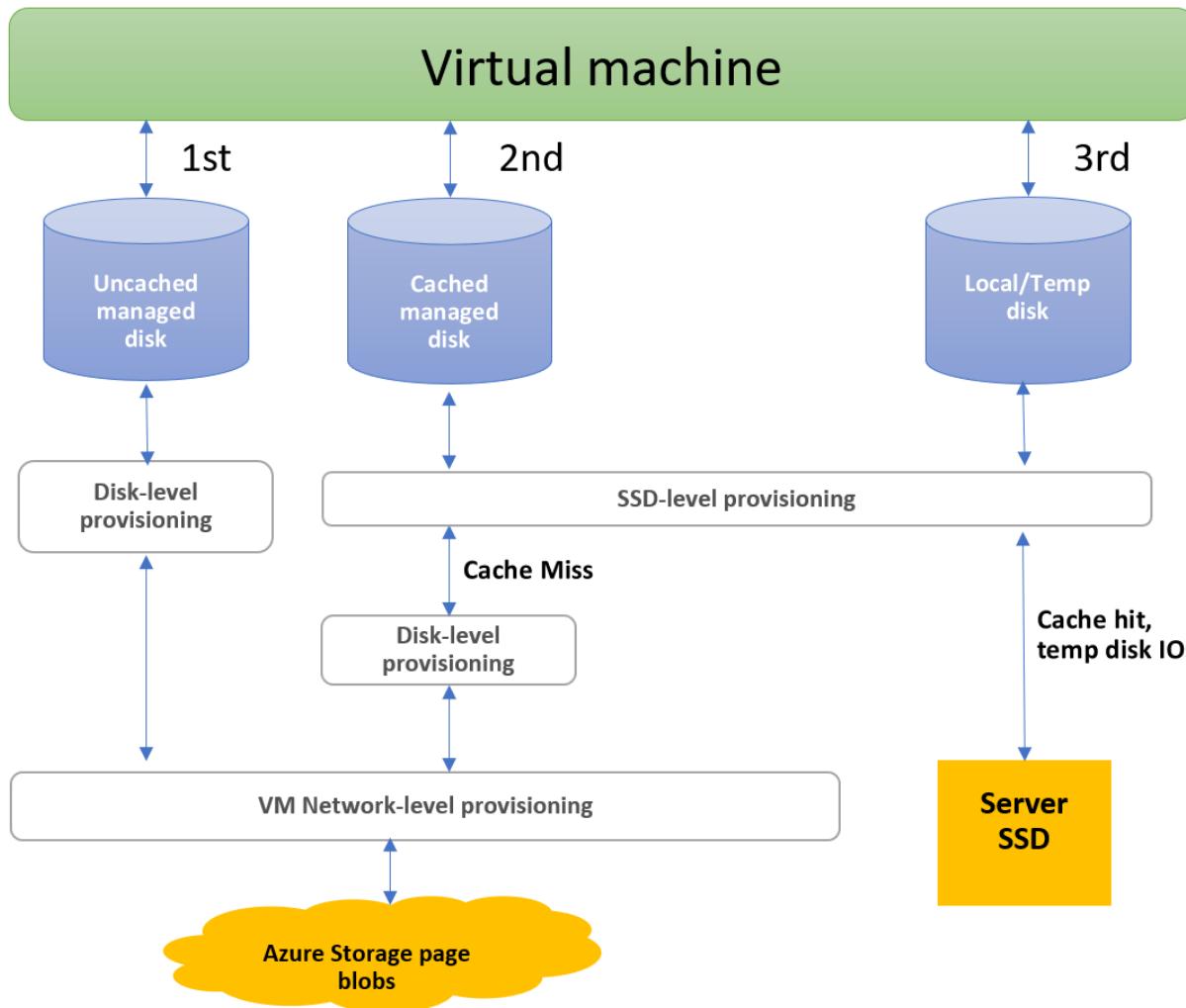
It's important to understand the difference between images and snapshots. With managed disks, you can take an image of a generalized VM that has been deallocated. This image includes all of the disks attached to the VM. You can use this image to create a VM, and it includes all of the disks.

A snapshot is a copy of a disk at the point in time the snapshot is taken. It applies only to one disk. If you have a VM that has one disk (the OS disk), you can take a snapshot or an image of it and create a VM from either the snapshot or the image.

A snapshot doesn't have awareness of any disk except the one it contains. This makes it problematic to use in scenarios that require the coordination of multiple disks, such as striping. Snapshots would need to be able to coordinate with each other and this is currently not supported.

## Disk allocation and performance

The following diagram depicts real-time allocation of bandwidth and IOPS for disks, with three different paths an IO can take:



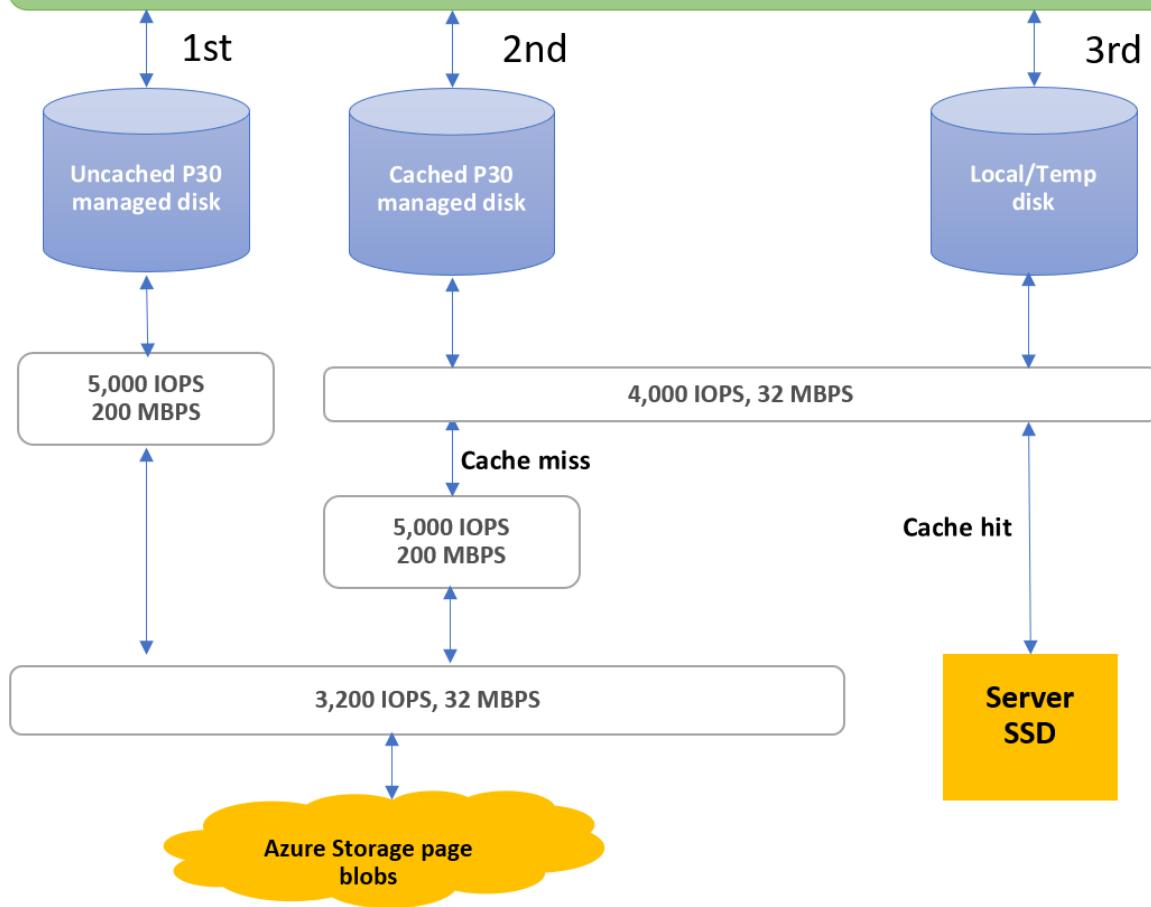
The first IO path is the uncached managed disk path. This path is taken if you are using a managed disk and set the host caching to none. An IO using this path will execute based on disk-level provisioning and then VM network-level provisioning for IOPs and throughput.

The second IO Path is the cached managed disk path. Cached managed disk IO uses an SSD close to the VM, which have their own IOPs and throughput provisioned, and is labeled SSD-level provisioning in the diagram. When a cached managed disk initiates a read, the request first checks to see if the data is in the server SSD. If the data isn't present, this creates a cached miss and the IO then executes based on SSD-level provisioning, disk-level provisioning and then VM network-level provisioning for IOPs and throughput. When the server SSD initiates reads on cached IO that are present on the server SSD, it creates a cache hit and the IO will then execute based on the SSD-level provisioning. Writes initiated by a cached managed disk always follow the path of a cached-miss, and need to go through SSD-level, disk-level, and VM network-level provisioning.

Finally, the third path is for the local/temp disk. This is available only on VMs that support local/temp disks. An IO using this path will execute based on SSD-Level Provisioning for IOPs and throughput.

As an example of these limitations, a Standard\_DS1v1 VM is prevented from achieving the 5,000 IOPS potential of a P30 disk, whether it is cached or not, because of limits at the SSD and network levels:

## Virtual machine: Standard\_DS1v1



Azure uses prioritized network channel for disk traffic, which gets the precedence over other low priority of network traffic. This helps disks maintain their expected performance in case of network contentions. Similarly, Azure Storage handles resource contentions and other issues in the background with automatic load balancing. Azure Storage allocates required resources when you create a disk, and applies proactive and reactive balancing of resources to handle the traffic level. This further ensures disks can sustain their expected IOPS and throughput targets. You can use the VM-level and Disk-level metrics to track the performance and setup alerts as needed.

Refer to our [design for high performance](#) article, to learn the best practices for optimizing VM + Disk configurations so that you can achieve your desired performance

## Next steps

If you'd like a video going into more detail on managed disks, check out: [Better Azure VM Resiliency with Managed Disks].

Learn more about the individual disk types Azure offers, which type is a good fit for your needs, and learn about their performance targets in our article on disk types.

[Select a disk type for IaaS VMs](#)

# Azure managed disk types

9/21/2022 • 18 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure managed disks currently offers five disk types, each intended to address a specific customer scenario:

- [Ultra disks](#)
- [Premium SSD v2 \(preview\)](#)
- [Premium SSDs \(solid-state drives\)](#)
- [Standard SSDs](#)
- [Standard HDDs \(hard disk drives\)](#)

## Disk type comparison

The following table provides a comparison of the five disk types to help you decide which to use.

	ULTRA DISK	PREMIUM SSD V2	PREMIUM SSD	STANDARD SSD	STANDARD HDD
Disk type	SSD	SSD	SSD	SSD	HDD
Scenario	IO-intensive workloads such as <a href="#">SAP HANA</a> , top tier databases (for example, SQL, Oracle), and other transaction-heavy workloads.	Production and performance-sensitive workloads that consistently require low latency and high IOPS and throughput	Production and performance sensitive workloads	Web servers, lightly used enterprise applications and dev/test	Backup, non-critical, infrequent access
Max disk size	65,536 gibibyte (GiB)	65,536 GiB	32,767 GiB	32,767 GiB	32,767 GiB
Max throughput	4,000 MB/s	1,200 MB/s	900 MB/s	750 MB/s	500 MB/s
Max IOPS	160,000	80,000	20,000	6,000	2,000
Usable as OS Disk?	No	No	Yes	Yes	Yes

## Ultra disks

Azure ultra disks are the highest-performing storage option for Azure virtual machines (VMs). You can change the performance parameters of an ultra disk without having to restart your VMs. Ultra disks are suited for data-intensive workloads such as SAP HANA, top-tier databases, and transaction-heavy workloads.

Ultra disks must be used as data disks and can only be created as empty disks. You should use Premium solid-state drives (SSDs) as operating system (OS) disks.

## Ultra disk size

Azure ultra disks offer up to 32-TiB per region per subscription by default, but ultra disks support higher capacity by request. To request an increase in capacity, request a quota increase or contact Azure Support.

The following table provides a comparison of disk sizes and performance caps to help you decide which to use.

DISK SIZE (GiB)	IOPS CAP	THROUGHPUT CAP (MBPS)
4	1,200	300
8	2,400	600
16	4,800	1,200
32	9,600	2,400
64	19,200	4,000
128	38,400	4,000
256	76,800	4,000
512	153,600	4,000
1,024-65,536 (sizes in this range increasing in increments of 1 TiB)	160,000	4,000

Ultra disks are designed to provide submillisecond latencies and target IOPS and throughput described in the preceding table 99.99% of the time.

## Ultra disk performance

Ultra disks feature a flexible performance configuration model that allows you to independently configure IOPS and throughput both before and after you provision the disk. Ultra disks come in several fixed sizes, ranging from 4 GiB up to 64 TiB.

### Ultra disk IOPS

Ultra disks support IOPS limits of 300 IOPS/GiB, up to a maximum of 160,000 IOPS per disk. To achieve the target IOPS for the disk, ensure that the selected disk IOPS are less than the VM IOPS limit.

The current maximum limit for IOPS for a single VM in generally available sizes is 80,000. Ultra disks with greater IOPS can be used as shared disks to support multiple VMs.

The minimum guaranteed IOPS per disk are 1 IOPS/GiB, with an overall baseline minimum of 100 IOPS. For example, if you provisioned a 4-GiB ultra disk, the minimum IOPS for that disk is 100, instead of four.

For more information about IOPS, see [Virtual machine and disk performance](#).

### Ultra disk throughput

The throughput limit of a single ultra disk is 256-KiB/s for each provisioned IOPS, up to a maximum of 4000 MBps per disk (where MBps =  $10^6$  Bytes per second). The minimum guaranteed throughput per disk is 4KiB/s for each provisioned IOPS, with an overall baseline minimum of 1 MBps.

You can adjust ultra disk IOPS and throughput performance at runtime without detaching the disk from the virtual machine. After a performance resize operation has been issued on a disk, it can take up to an hour for the change to take effect. Up to four performance resize operations are permitted during a 24-hour window.

It's possible for a performance resize operation to fail because of a lack of performance bandwidth capacity.

## Ultra disk limitations

Ultra disks can't be used as OS disks, they can only be created as empty data disks. Ultra disks also can't be used with some features and functionality, including disk snapshots, disk export, changing disk type, VM images, availability sets, Azure Dedicated Hosts, or Azure disk encryption. Azure Backup and Azure Site Recovery do not support ultra disks. In addition, only un-cached reads and un-cached writes are supported.

Ultra disks support a 4k physical sector size by default. A 512E sector size is available as a generally available offering with no sign-up required. While most applications are compatible with 4k sector sizes, some require 512 byte sector sizes. Oracle Database, for example, requires release 12.2 or later in order to support 4k native disks. For older versions of Oracle DB, 512 byte sector size is required.

The only infrastructure redundancy options currently available to ultra disks are availability zones. VMs using any other redundancy options cannot attach an ultra disk.

The following table outlines the regions ultra disks are available in, as well as their corresponding availability options.

### NOTE

If a region in the following list lacks availability zones that support ultra disks, then a VM in that region must be deployed without infrastructure redundancy in order to attach an ultra disk.

REDUNDANCY OPTIONS	REGIONS
Single VMs	Australia Central Brazil South Central India East Asia Germany West Central Korea Central North Central US, South Central US, West US US Gov Arizona, US Gov Texas, US Gov Virginia
One availability zone	China North 3 Qatar Central
Two availability zones	France Central
Three availability zones	Australia East Canada Central North Europe, West Europe Japan East Southeast Asia Sweden Central UK South Central US, East US, East US 2, West US 2, West US 3

Not every VM size is available in every supported region with ultra disks. The following table lists VM series which are compatible with ultra disks.

VM TYPE	SIZES	DESCRIPTION
---------	-------	-------------

VM TYPE	SIZES	DESCRIPTION
General purpose	<a href="#">DSv3-series</a> , <a href="#">Ddsv4-series</a> , <a href="#">Dsv4-series</a> , <a href="#">Dasv4-series</a> , <a href="#">Dsv5-series</a> , <a href="#">Ddsv5-series</a> , <a href="#">Dasv5-series</a>	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	<a href="#">FSv2-series</a>	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	<a href="#">ESv3-series</a> , <a href="#">Easv4-series</a> , <a href="#">Edsv4-series</a> , <a href="#">Esv4-series</a> , <a href="#">Esv5-series</a> , <a href="#">Edsv5-series</a> , <a href="#">Easv5-series</a> , <a href="#">Ebsv5 series</a> , <a href="#">Ebdsv5 series</a> , <a href="#">M-series</a> , <a href="#">Mv2-series</a> , <a href="#">Msv2/Mdsv2-series</a>	High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	<a href="#">LSv2-series</a>	High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases.
GPU optimized	<a href="#">NCv2-series</a> , <a href="#">NCv3-series</a> , <a href="#">NCasT4_v3-series</a> , <a href="#">ND-series</a> , <a href="#">NDv2-series</a> , <a href="#">NVv3-series</a> , <a href="#">NVv4-series</a> , <a href="#">NVadsA10 v5-series</a>	Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs.
Performance optimized	<a href="#">HB-series</a> , <a href="#">HC-series</a> , <a href="#">HBv2-series</a>	The fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

If you would like to start using ultra disks, see the article on [using Azure ultra disks](#).

## Premium SSD v2 (preview)

Azure Premium SSD v2 (preview) is designed for IO-intense enterprise workloads that require consistent sub-millisecond disk latencies and high IOPS and throughput at a low cost. The performance (capacity, throughput, and IOPS) of Premium SSD v2 disks can be independently configured at any time, making it easier for more scenarios to be cost efficient while meeting performance needs. For example, a transaction-intensive database workload may need a large amount of IOPS at a small size, or a gaming application may need a large amount of IOPS during peak hours. Premium SSD v2 is suited for a broad range of workloads such as SQL server, Oracle, MariaDB, SAP, Cassandra, Mongo DB, big data/analytics, and gaming, on virtual machines or stateful containers.

### Differences between Premium SSD and Premium SSD v2

Unlike Premium SSDs, Premium SSD v2 doesn't have dedicated sizes. You can set a Premium SSD v2 to any supported size you prefer, and make granular adjustments to the performance without downtime. Premium SSD v2 doesn't support host caching but, benefits significantly from lower latency which addresses some of the same core problems host caching addresses. The ability to adjust IOPS, throughput, and size at any time also means you can avoid the maintenance overhead of having to stripe disks to meet your needs.

### Premium SSD v2 limitations

- Premium SSD v2 disks can't be used as an OS disk.
- Currently, Premium SSD v2 disks can only be attached to zonal VMs.

- Currently, Premium SSD v2 disks can't be attached to VMs in virtual machine scale sets.
- Currently, taking snapshots aren't supported, and you can't create a Premium SSD v2 from the snapshot of another disk type.
- Currently, Premium SSD v2 disks can't be attached to VMs with encryption at host enabled.
- Currently, Premium SSD v2 disks can't be attached to VMs in Availability Sets.
- Azure Disk Encryption isn't supported for VMs with Premium SSD v2 disks.
- Azure Backup and Azure Site Recovery aren't supported for VMs with Premium SSD v2 disks.

#### **Regional availability**

Currently only available in the following regions:

- US East
- West Europe

#### **Premium SSD v2 performance**

With Premium SSD v2 disks, you can individually set the capacity, throughput, and IOPS of a disk based on your workload needs, providing you more flexibility and reduced costs. Each of these values determine the cost of your disk.

#### **Premium SSD v2 capacities**

Premium SSD v2 capacities range from 1 GiB to 64 TiBs, in 1-GiB increments. You're billed on a per GiB ratio, see the [pricing page](#) for details.

Premium SSD v2 offers up to 32 TiBs per region per subscription by default in the public preview, but supports higher capacity by request. To request an increase in capacity, request a quota increase or contact Azure Support.

#### **Premium SSD v2 IOPS**

All Premium SSD v2 disks have a baseline IOPS of 3000 that is free of charge. After 6 GiB, the maximum IOPS a disk can have increases at a rate of 500 per GiB, up to 80,000 IOPS. So an 8 GiB disk can have up to 4,000 IOPS, and a 10 GiB can have up to 5,000 IOPS. To be able to set 80,000 IOPS on a disk, that disk must have at least 160 GiBs. Increasing your IOPS beyond 3000 increases the price of your disk.

#### **Premium SSD v2 throughput**

All Premium SSD v2 disks have a baseline throughput of 125 MB/s, that is free of charge. After 6 GiB, the maximum throughput that can be set increases by 0.25 MB/s per set IOPS. If a disk has 3,000 IOPS, the max throughput it can set is 750 MB/s. To raise the throughput for this disk beyond 750 MB/s, its IOPS must be increased. For example, if you increased the IOPS to 4,000, then the max throughput that can be set is 1,000. 1,200 MB/s is the maximum throughput supported for disks that have 5,000 IOPS or more. Increasing your throughput beyond 125 increases the price of your disk.

#### **Premium SSD v2 Sector Sizes**

Premium SSD v2 supports a 4k physical sector size by default. A 512E sector size is also supported. While most applications are compatible with 4k sector sizes, some require 512-byte sector sizes. Oracle Database, for example, requires release 12.2 or later in order to support 4k native disks. For older versions of Oracle DB, 512-byte sector size is required.

#### **Summary**

The following table provides a comparison of disk capacities and performance maximums to help you decide which to use.

DISK SIZE	MAXIMUM AVAILABLE IOPS	MAXIMUM AVAILABLE THROUGHPUT (MB/S)
1 GiB-64 TiBs	3,000-80,000 (Increases by 500 IOPS per GiB)	125-1,200 (increases by 0.25 MB/s per set IOPS)

To deploy a Premium SSD v2, see [Deploy a Premium SSD v2 \(preview\)](#).

## Premium SSDs

Azure Premium SSDs deliver high-performance and low-latency disk support for virtual machines (VMs) with input/output (IO)-intensive workloads. To take advantage of the speed and performance of Premium SSDs, you can migrate existing VM disks to Premium SSDs. Premium SSDs are suitable for mission-critical production applications, but you can use them only with compatible VM series.

To learn more about individual Azure VM types and sizes for Windows or Linux, including size compatibility for premium storage, see [Sizes for virtual machines in Azure](#). You'll need to check each individual VM size article to determine if it's premium storage-compatible.

### Premium SSD size

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Disk size in GiB	4	8	16	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
Provisioned IOPS per disk	120	120	120	120	240	500	1,100	2,300	5,000	7,500	7,500	16,000	18,000	20,000
Provisioned Throughput per disk	25 MB/sec	25 MB/sec	25 MB/sec	25 MB/sec	50 MB/sec	100 MB/sec	125 MB/sec	150 MB/sec	200 MB/sec	250 MB/sec	250 MB/sec	500 MB/sec	750 MB/sec	900 MB/sec
Max burst IOPS per disk	3,500	3,500	3,500	3,500	3,500	3,500	3,500	3,500	30,000*	30,000*	30,000*	30,000*	30,000*	30,000*

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Max burst throughput per disk	170 MB/sec	1,000 MB/sec*												
Max burst duration	30 min	Unlimited*	Unlimited*	Unlimited*	Unlimited*	Unlimited*	Unlimited*							
Eligible for reservation	No	Yes, up to one year												

\*Applies only to disks with on-demand bursting enabled.

Capacity, IOPS, and throughput are guaranteed when a premium storage disk is provisioned. For example, if you create a P50 disk, Azure provisions 4,095-GB storage capacity, 7,500 IOPS, and 250-MB/s throughput for that disk. Your application can use all or part of the capacity and performance. Premium SSDs are designed to provide the single-digit millisecond latencies, target IOPS, and throughput described in the preceding table 99.9% of the time.

### Premium SSD bursting

Premium SSDs offer disk bursting, which provides better tolerance on unpredictable changes of IO patterns. Disk bursting is especially useful during OS disk boot and for applications with spiky traffic. To learn more about how bursting for Azure disks works, see [Disk-level bursting](#).

### Premium SSD transactions

For Premium SSDs, each I/O operation less than or equal to 256 KiB of throughput is considered a single I/O operation. I/O operations larger than 256 KiB of throughput are considered multiple I/Os of size 256 KiB.

## Standard SSDs

Azure standard SSDs are optimized for workloads that need consistent performance at lower IOPS levels. They're an especially good choice for customers with varying workloads supported by on-premises hard disk drive (HDD) solutions. Compared to standard HDDs, standard SSDs deliver better availability, consistency, reliability, and latency. Standard SSDs are suitable for web servers, low IOPS application servers, lightly used enterprise applications, and non-production workloads. Like standard HDDs, standard SSDs are available on all Azure VMs.

## Standard SSD size

STAN NDAR DSSD SIZES	E1	E2	E3	E4	E6	E10	E15	E20	E30	E40	E50	E60	E70	E80
Disk size in GiB	4	8	16	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
IOPS per disk	Up to 500	Up to 2,000	Up to 4,000	Up to 6,000										
Throughput per disk	Up to 60 MB/sec	Up to 400 MB/sec	Up to 600 MB/sec	Up to 750 MB/sec										
Max burst IOPS per disk	600	600	600	600	600	600	600	600	1000					
Max burst throughput per disk	150 MB/sec	250 MB/sec												
Max burst duration	30 min													

Standard SSDs are designed to provide single-digit millisecond latencies and the IOPS and throughput up to the limits described in the preceding table 99% of the time. Actual IOPS and throughput may vary sometimes depending on the traffic patterns. Standard SSDs will provide more consistent performance than the HDD disks with the lower latency.

## Standard SSD transactions

For standard SSDs, each I/O operation less than or equal to 256 KiB of throughput is considered a single I/O operation. I/O operations larger than 256 KiB of throughput are considered multiple I/Os of size 256 KiB. These transactions incur a billable cost.

### Standard SSD Bursting

Standard SSDs offer disk bursting, which provides better tolerance for the unpredictable IO pattern changes. OS boot disks and applications prone to traffic spikes will both benefit from disk bursting. To learn more about how bursting for Azure disks works, see [Disk-level bursting](#).

## Standard HDDs

Azure standard HDDs deliver reliable, low-cost disk support for VMs running latency-tolerant workloads. With standard storage, your data is stored on HDDs, and performance may vary more widely than that of SSD-based disks. Standard HDDs are designed to deliver write latencies of less than 10 ms and read latencies of less than 20 ms for most IO operations. Actual performance may vary depending on IO size and workload pattern, however. When working with VMs, you can use standard HDD disks for dev/test scenarios and less critical workloads. Standard HDDs are available in all Azure regions and can be used with all Azure VMs.

### Standard HDD size

STANDARD DISK TYPE	S4	S6	S10	S15	S20	S30	S40	S50	S60	S70	S80
Disk size in GiB	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
IOPS per disk	Up to 500	Up to 1,300	Up to 2,000	Up to 2,000							
Throughput per disk	Up to 60 MB/sec	Up to 300 MB/sec	Up to 500 MB/sec	Up to 500 MB/sec							

### Standard HDD Transactions

For Standard HDDs, each I/O operation is considered as a single transaction, whatever the I/O size. These transactions have a billing impact.

## Billing

When using managed disks, the following billing considerations apply:

- Disk type
- Managed disk Size
- Snapshots
- Outbound data transfers
- Number of transactions

**Managed disk size:** Managed disks are billed according to their provisioned size. Azure maps the provisioned size (rounded up) to the nearest offered disk size. For details of the disk sizes offered, see the previous tables. Each disk maps to a supported provisioned disk-size offering and is billed accordingly. For example, if you provisioned a 200-GiB standard SSD, it maps to the disk size offer of E15 (256 GiB). Billing for any provisioned

disk is prorated hourly by using the monthly price for the storage offering. For example, you provision an E10 disk and delete it after 20 hours of use. In this case, you're billed for the E10 offering prorated to 20 hours, regardless of the amount of data written to the disk.

**Snapshots:** Snapshots are billed based on the size used. For example, you create a snapshot of a managed disk with provisioned capacity of 64 GiB and actual used data size of 10 GiB. In this case, the snapshot is billed only for the used data size of 10 GiB.

For more information on snapshots, see the section on snapshots in the [managed disk overview](#).

**Outbound data transfers:** [Outbound data transfers](#) (data going out of Azure data centers) incur billing for bandwidth usage.

**Transactions:** You're billed for the number of transactions performed on a standard managed disk. For standard SSDs, each I/O operation less than or equal to 256 KiB of throughput is considered a single I/O operation. I/O operations larger than 256 KiB of throughput are considered multiple I/Os of size 256 KiB. For Standard HDDs, each IO operation is considered a single transaction, whatever the I/O size.

For detailed information on pricing for managed disks (including transaction costs), see [Managed Disks Pricing](#).

### Ultra disks VM reservation fee

Azure VMs have the capability to indicate if they're compatible with ultra disks. An ultra disk-compatible VM allocates dedicated bandwidth capacity between the compute VM instance and the block storage scale unit to optimize the performance and reduce latency. When you add this capability on the VM, it results in a reservation charge. The reservation charge is only imposed if you enabled ultra disk capability on the VM without an attached ultra disk. When an ultra disk is attached to the ultra disk compatible VM, the reservation charge wouldn't be applied. This charge is per vCPU provisioned on the VM.

#### NOTE

For [constrained core VM sizes](#), the reservation fee is based on the actual number of vCPUs and not the constrained cores. For Standard\_E32-8s\_v3, the reservation fee will be based on 32 cores.

Refer to the [Azure Disks pricing page](#) for ultra disk pricing details.

### Azure disk reservation

Disk reservation provides you a discount on the advance purchase of one year's of disk storage, reducing your total cost. When you purchase a disk reservation, you select a specific disk SKU in a target region. For example, you may choose five P30 (1 TiB) Premium SSDs in the Central US region for a one year term. The disk reservation experience is similar to Azure reserved VM instances. You can bundle VM and Disk reservations to maximize your savings. For now, Azure Disks Reservation offers one year commitment plan for Premium SSD SKUs from P30 (1 TiB) to P80 (32 TiB) in all production regions. For more information about reserved disks pricing, see [Azure Disks pricing page](#).

## Next steps

See [Managed Disks pricing](#) to get started.

# Redundancy options for managed disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure managed disks offer two storage redundancy options, zone-redundant storage (ZRS), and locally-redundant storage. ZRS provides higher availability for managed disks than locally-redundant storage (LRS) does. However, the write latency for LRS disks is better than ZRS disks because LRS disks synchronously write data to three copies in a single data center.

## Locally-redundant storage for managed disks

Locally-redundant storage (LRS) replicates your data three times within a single data center in the selected region. LRS protects your data against server rack and drive failures. To protect an LRS disk from a zonal failure like a natural disaster or other issues, take the following steps:

- Use applications that can synchronously write data to two zones, and automatically failover to another zone during a disaster.
  - An example would be SQL Server Always On.
- Take frequent backups of LRS disks with ZRS snapshots.
- Enable cross-zone disaster recovery for LRS disks via [Azure Site Recovery](#). However, cross-zone disaster recovery doesn't provide zero Recovery Point Objective (RPO).

If your workflow doesn't support application-level synchronous writes across zones, or your application must meet zero RPO, then ZRS disks would ideal.

## Zone-redundant storage for managed disks

Zone-redundant storage (ZRS) synchronously replicates your Azure managed disk across three Azure availability zones in the region you select. Each availability zone is a separate physical location with independent power, cooling, and networking.

A ZRS disk lets you recover from failures in availability zones. If a zone went down, a ZRS disk can be attached to a virtual machine (VM) in a different zone. ZRS disks can also be shared between VMs for improved availability with clustered or distributed applications like SQL FCI, SAP ASCS/SCS, or GFS2. A shared ZRS disk can be attached to primary and secondary VMs in different zones to take advantage of both ZRS and [availability zones](#). If your primary zone fails, you can quickly fail over to the secondary VM using [SCSI persistent reservation](#).

For more information on ZRS disks, see [Zone Redundant Storage \(ZRS\) option for Azure Disks for high availability](#).

### Billing implications

For details see the [Azure pricing page](#).

### Comparison with other disk types

Except for more write latency, disks using ZRS are identical to disks using LRS, they have the same scale targets. [Benchmark your disks](#) to simulate the workload of your application and compare the latency between LRS and ZRS disks.

### Limitations

ZRS for managed disks have the following restrictions:

- Only supported with premium solid-state drives (SSD) and standard SSDs.
- Currently available only in the West US 2, West Europe, North Europe, and France Central regions.

## Next steps

- To learn how to create a ZRS disk, see [Deploy a ZRS managed disk](#).

# Deploy a managed disk that uses zone-redundant storage

9/21/2022 • 6 minutes to read • [Edit Online](#)

This article covers how to deploy a disk that uses zone-redundant storage (ZRS) as a redundancy option. ZRS replicates your Azure managed disk synchronously across three Azure availability zones in the selected region. Each availability zone is a separate physical location with independent power, cooling, and networking.

For conceptual information on ZRS, see [Zone-redundant storage for managed disks](#)

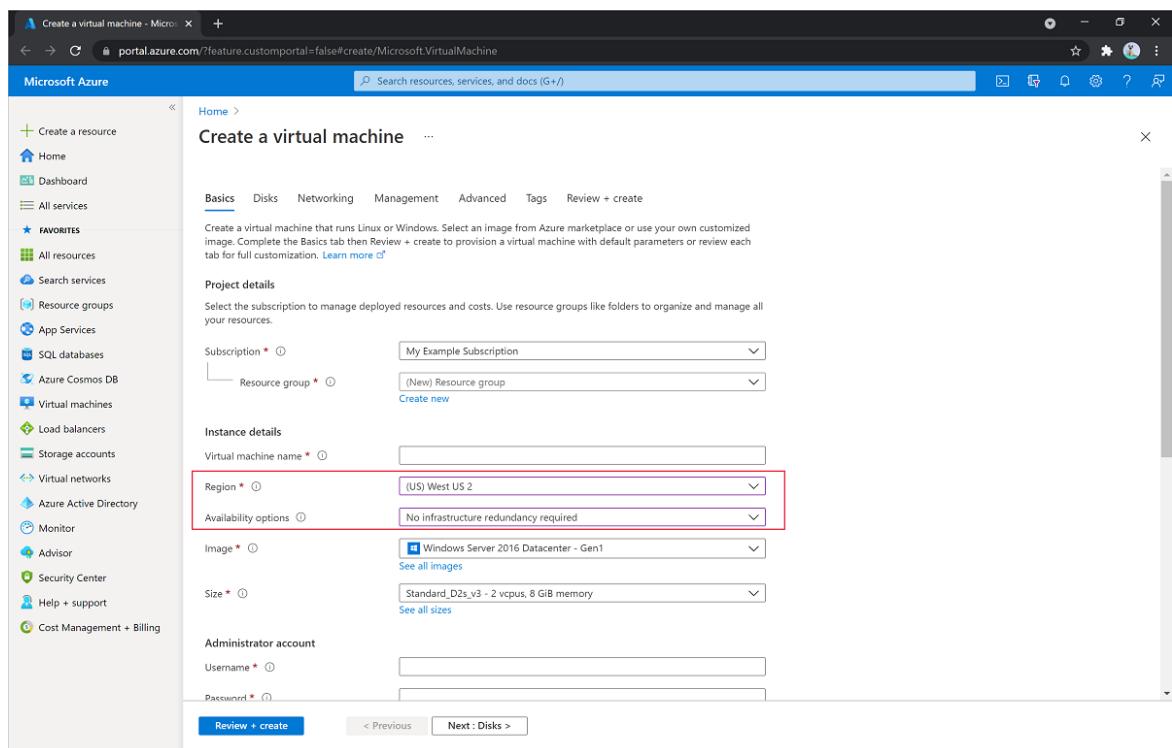
## Limitations

ZRS for managed disks have the following restrictions:

- Only supported with premium solid-state drives (SSD) and standard SSDs.
- Currently available only in the West US 2, West Europe, North Europe, and France Central regions.
- [Azure portal](#)
- [Azure CLI](#)
- [Azure PowerShell](#)
- [Resource Manager Template](#)

## Create a VM with a ZRS OS disk

1. Sign in to the [Azure portal](#).
2. Navigate to **Virtual machines** and follow the normal VM creation process.
3. Select a supported region and set **Availability options** to **No infrastructure redundancy required**.



4. Proceed to the **Disks** pane.

- Select your disk and select one of the ZRS disks in the drop-down.

**Create a virtual machine**

Basics Disks Networking Management Advanced Tags Review + create

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

**Disk options**

OS disk type *	Premium SSD (locally-redundant storage)
Encryption type *	Locally-redundant storage (data is replicated within a single datacenter)
Enable Ultra Disk compatibility	Premium SSD Best for production and performance sensitive workloads
Data disks	Standard SSD Best for web servers, lightly used enterprise applications and dev/test
You can add and configure additional data disks or temporary disk.	Standard HDD Best for backup, non-critical, and infrequent access
LUN Name	Zone-redundant storage (data is replicated within multiple datacenters)
Create and attach a new disk	Premium SSD Best for the production workloads that need storage resiliency against zone failures
Advanced	Standard SSD Best for web servers, lightly used enterprise applications and dev/test that need storage resiliency against zone failures

- Proceed through the rest of the VM deployment, making any choices that you desire.

You've now deployed a VM with a ZRS OS disk.

### Create a ZRS disk

- In the Azure portal, search for and select **Disks**.
- Select **+ Add** to create a new disk.
- Select a supported region and **Availability zone** to **None**.
- Select **Change size**.

Subscription *	My Example Subscription
Resource group *	<input type="text"/> Create new
<b>Disk details</b>	
Disk name *	<input type="text"/>
Region *	(US) West US 2
Availability zone	None
Source type	None
Size *	1024 GiB Premium SSD LRS <a href="#">Change size</a>

- Select one of the available ZRS disks and select **OK**.

Disk SKU ⓘ

Premium SSD (locally-redundant storage) ^

Locally-redundant storage (data is replicated within a single datacenter)

Premium SSD

Best for production and performance sensitive workloads

Standard SSD

Best for web servers, lightly used enterprise applications and dev/test

Standard HDD

Best for backup, non-critical, and infrequent access

Zone-redundant storage (data is replicated within multiple datacenters)

Premium SSD

Best for the production workloads that need storage resiliency against zone failures

Standard SSD

Best for web servers, lightly used enterprise applications and dev/test that need storage resiliency against zone failures

6. Continue through the deployment process.

You have now created a managed disk that uses ZRS.

## Next steps

- Check out more [Azure Resource Manager templates to create a VM with ZRS disks](#).

# Share an Azure managed disk

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure shared disks is a feature for Azure managed disks that allow you to attach a managed disk to multiple virtual machines (VMs) simultaneously. Attaching a managed disk to multiple VMs allows you to either deploy new or migrate existing clustered applications to Azure.

## How it works

VMs in the cluster can read or write to their attached disk based on the reservation chosen by the clustered application using [SCSI Persistent Reservations](#) (SCSI PR). SCSI PR is an industry standard used by applications running on Storage Area Network (SAN) on-premises. Enabling SCSI PR on a managed disk allows you to migrate these applications to Azure as-is.

Shared managed disks offer shared block storage that can be accessed from multiple VMs, these are exposed as logical unit numbers (LUNs). LUNs are then presented to an initiator (VM) from a target (disk). These LUNs look like direct-attached-storage (DAS) or a local drive to the VM.

Shared managed disks don't natively offer a fully managed file system that can be accessed using SMB/NFS. You need to use a cluster manager, like Windows Server Failover Cluster (WSFC), or Pacemaker, that handles cluster node communication and write locking.

## Limitations

### General limitations

Enabling shared disks is only available to a subset of disk types. Currently only ultra disks, premium SSD v2 (preview), premium SSDs, and standard SSDs can enable shared disks. Shared disks can be attached to individual VMSS instances but can't be defined in the VMSS models or automatically deployed.

Each managed disk that has shared disks enabled are also subject to the following limitations, organized by disk type:

### Ultra disks

Ultra disks have their own separate list of limitations, unrelated to shared disks. For ultra disk limitations, refer to [Using Azure ultra disks](#).

When sharing ultra disks, they have the following additional limitations:

- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Can't be shared across availability zones.

### Premium SSD v2 (preview)

Premium SSD v2 disks have their own separate list of limitations, unrelated to shared disks. For these limitations, see [Premium SSD v2 limitations](#).

When sharing Premium SSD v2 disks, they have the following additional limitation:

- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).

- Can't be shared across availability zones.

## Premium SSD

- Can only be enabled on data disks, not OS disks.
- Host caching isn't available for premium SSD disks with `maxShares>1`.
- Disk bursting isn't available for premium SSD disks with `maxShares>1`.
- When using Availability sets or virtual machine scale sets with Azure shared disks, [storage fault domain alignment](#) with virtual machine fault domain isn't enforced for the shared data disk.
- When using [proximity placement groups \(PPG\)](#), all virtual machines sharing a disk must be part of the same PPG.
- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Azure Site Recovery support isn't yet available.
- Azure Backup is available through [Azure Disk Backup](#).
- Only [server-side encryption](#) is supported, [Azure Disk Encryption](#) isn't currently supported.
- Can only be shared across availability zones if using [Zone-redundant storage for managed disks](#).

## Standard SSDs

- Can only be enabled on data disks, not OS disks.
- Host caching isn't available for standard SSDs with `maxShares>1`.
- When using Availability sets and virtual machine scale sets with Azure shared disks, [storage fault domain alignment](#) with virtual machine fault domain isn't enforced for the shared data disk.
- When using [proximity placement groups \(PPG\)](#), all virtual machines sharing a disk must be part of the same PPG.
- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Azure Site Recovery support isn't yet available.
- Azure Backup is available through [Azure Disk Backup](#).
- Only [server-side encryption](#) is supported, [Azure Disk Encryption](#) isn't currently supported.
- Can only be shared across availability zones if using [Zone-redundant storage for managed disks](#).

## Operating system requirements

Shared disks support several operating systems. See the [Windows](#) or [Linux](#) sections for the supported operating systems.

## Billing implications

When you share a disk, your billing could be impacted in two different ways, depending on the type of disk.

For shared premium SSD disks, in addition to cost of the disk's tier, there's an extra charge that increases with each VM the SSD is mounted to. See [managed disks pricing](#) for details.

Ultra disks don't have an extra charge for each VM that they're mounted to. They're billed on the total IOPS and MBps that the disk is configured for. Normally, an ultra disk has two performance throttles that determine its total IOPS/MBps. However, when configured as a shared ultra disk, two more performance throttles are exposed, for a total of four. These two additional throttles allow for increased performance at an extra expense and each meter has a default value, which raises the performance and cost of the disk.

The four performance throttles a shared ultra disk has are `diskMBpsReadWrite`, `diskIOPSReadOnly`, `diskMBpsReadWrite`, and `diskMBpsReadOnly`. Each performance throttle can be configured to change the performance of your disk. The performance for shared ultra disk is calculated in the following ways: total provisioned IOPS (`diskIOPSReadWrite + diskIOPSReadOnly`) and for total provisioned throughput MBps

(diskMBpsReadWrite + diskMBpsReadOnly).

Once you've determined your total provisioned IOPS and total provisioned throughput, you can use them in the [pricing calculator](#) to determine the cost of an ultra shared disk.

## Disk sizes

For now, only ultra disks, premium SSD v2 (preview), premium SSD, and standard SSDs can enable shared disks. Different disk sizes may have a different `maxShares` limit, which you can't exceed when setting the `maxShares` value.

For each disk, you can define a `maxShares` value that represents the maximum number of nodes that can simultaneously share the disk. For example, if you plan to set up a 2-node failover cluster, you would set `maxShares=2`. The maximum value is an upper bound. Nodes can join or leave the cluster (mount or unmount the disk) as long as the number of nodes is lower than the specified `maxShares` value.

### NOTE

The `maxShares` value can only be set or edited when the disk is detached from all nodes.

### Premium SSD ranges

The following table illustrates the allowed maximum values for `maxShares` by premium SSD sizes:

DISK SIZES	MAXSHARES LIMIT
P1,P2,P3,P4,P6,P10,P15,P20	3
P30, P40, P50	5
P60, P70, P80	10

The IOPS and bandwidth limits for a disk aren't affected by the `maxShares` value. For example, the max IOPS of a P15 disk is 1100 whether `maxShares = 1` or `maxShares > 1`.

### Standard SSD ranges

The following table illustrates the allowed maximum values for `maxShares` by standard SSD sizes:

DISK SIZES	MAXSHARES LIMIT
E1,E2,E3,E4,E6,E10,E15,E20	3
E30, E40, E50	5
E60, E70, E80	10

The IOPS and bandwidth limits for a disk aren't affected by the `maxShares` value. For example, the max IOPS of a E15 disk is 500 whether `maxShares = 1` or `maxShares > 1`.

### Ultra disk ranges

The minimum `maxShares` value is 1, while the maximum `maxShares` value is 15. There are no size restrictions on ultra disks, any size ultra disk can use any value for `maxShares`, up to and including the maximum value.

### Premium SSD v2 ranges

The minimum `maxShares` value is 1, while the maximum `maxShares` value is 15. There are no size restrictions on

Premium SSD v2, any size Premium SSD v2 disk can use any value for `maxShares`, up to and including the maximum value.

## Sample workloads

### Windows

Azure shared disks are supported on Windows Server 2008 and newer. Most Windows-based clustering builds on WSFC, which handles all core infrastructure for cluster node communication, allowing your applications to take advantage of parallel access patterns. WSFC enables both CSV and non-CSV-based options depending on your version of Windows Server. For details, refer to [Create a failover cluster](#).

Some popular applications running on WSFC include:

- [Create an FCI with Azure shared disks \(SQL Server on Azure VMs\)](#)
  - [Migrate your failover cluster instance to SQL Server on Azure VMs with shared disks](#)
- Scale-out File Server (SoFS) [template](#)
- SAP ASCS/SCS [template](#)
- File Server for General Use (IW workload)
- Remote Desktop Server User Profile Disk (RDS UPD)

### Linux

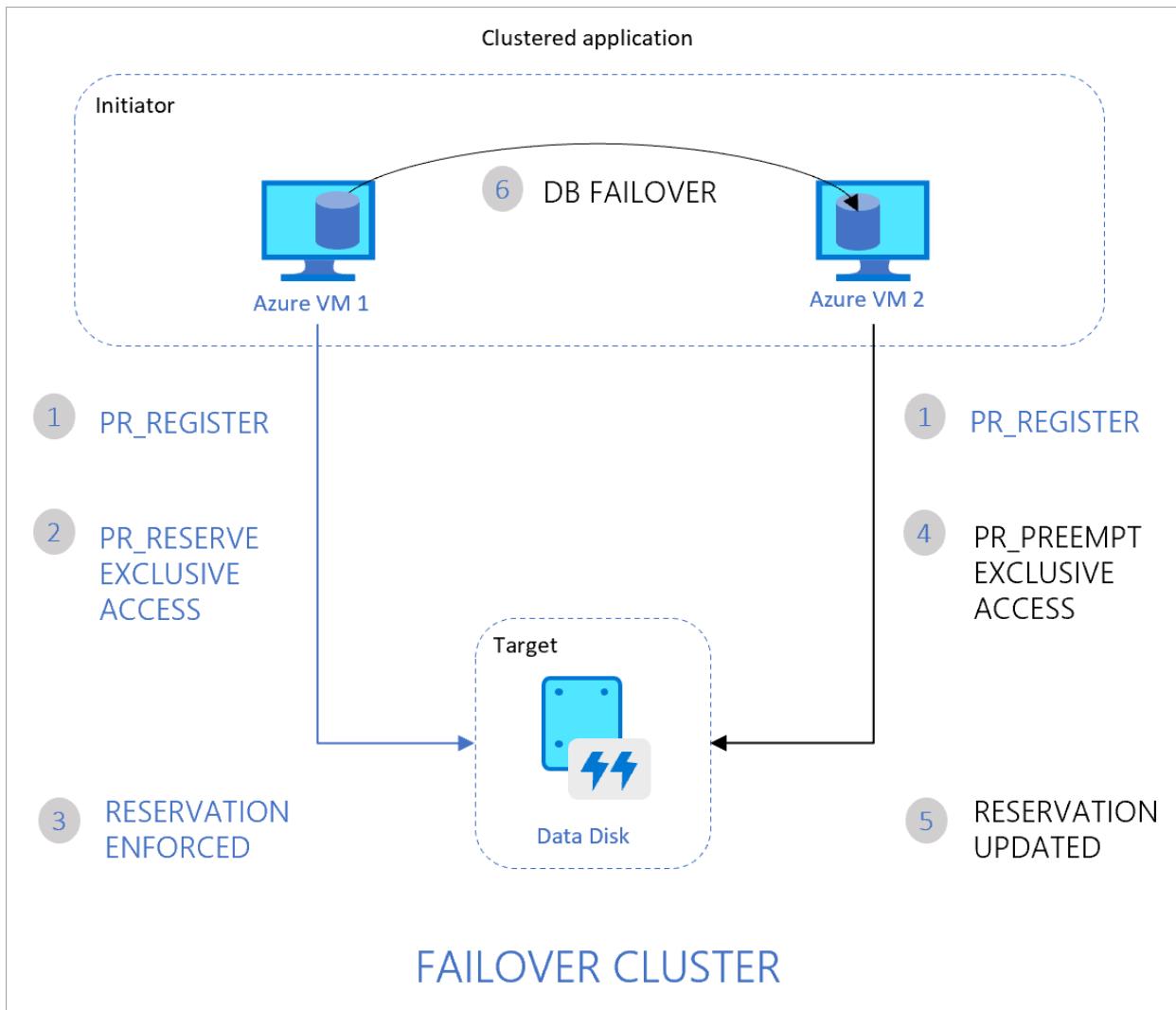
Azure shared disks are supported on:

- [SUSE SLE HA 15 SP1 and above](#)
- [Ubuntu 18.04 and above](#)
- Red Hat Enterprise Linux (RHEL) ([support policy](#))
  - [RHEL 7.9](#)
  - [RHEL 8.3 and above](#)
- [Oracle Enterprise Linux](#)

Linux clusters can use cluster managers such as [Pacemaker](#). Pacemaker builds on [Corosync](#), enabling cluster communications for applications deployed in highly available environments. Some common clustered filesystems include [ocfs2](#) and [gfs2](#). You can use SCSI Persistent Reservation (SCSI PR) and/or STONITH Block Device (SBD) based clustering models for arbitrating access to the disk. When using SCSI PR, you can manipulate reservations and registrations using utilities such as [fence\\_scsi](#) and [sg\\_persist](#).

## Persistent reservation flow

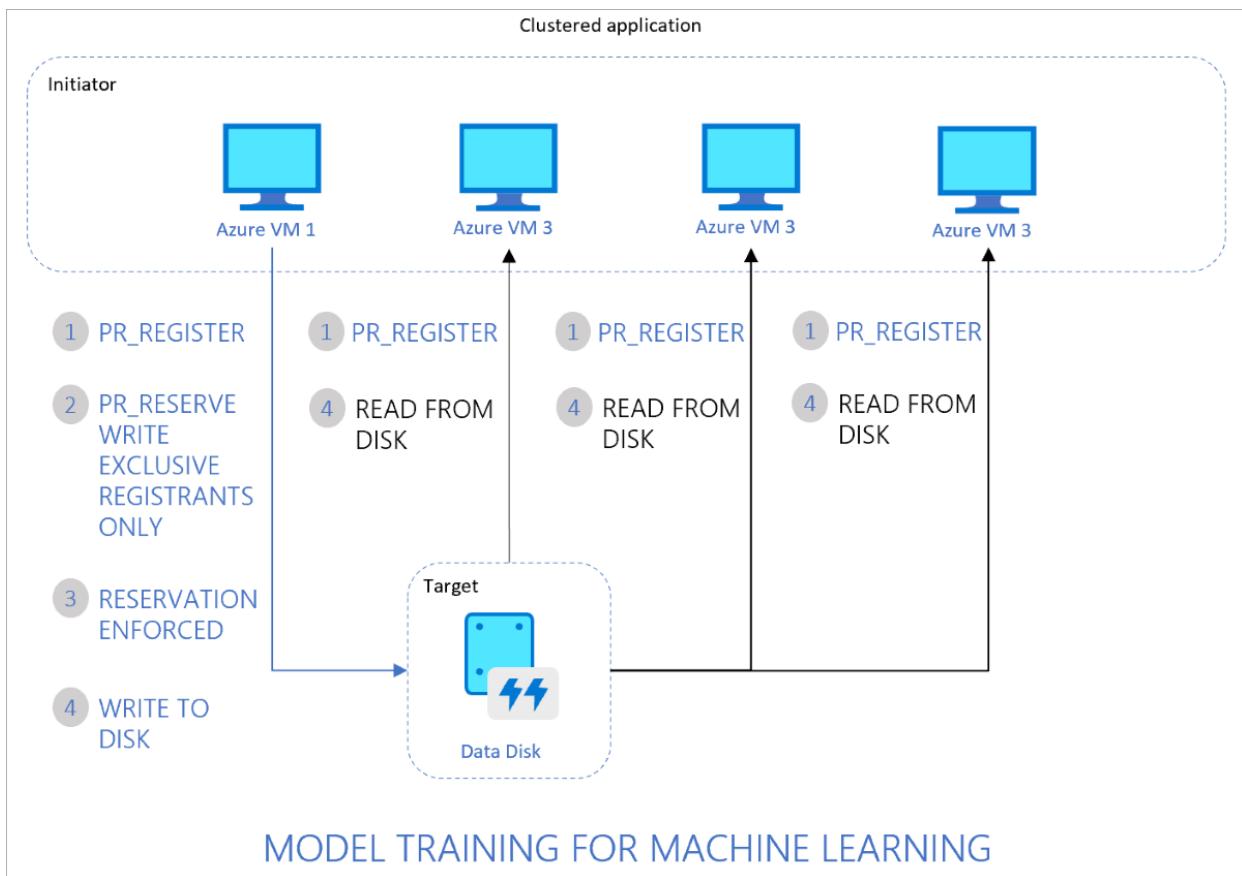
The following diagram illustrates a sample 2-node clustered database application that uses SCSI PR to enable failover from one node to the other.



The flow is as follows:

1. The clustered application running on both Azure VM1 and VM2 registers its intent to read or write to the disk.
2. The application instance on VM1 then takes exclusive reservation to write to the disk.
3. This reservation is enforced on your Azure disk and the database can now exclusively write to the disk. Any writes from the application instance on VM2 won't succeed.
4. If the application instance on VM1 goes down, the instance on VM2 can now initiate a database failover and take-over of the disk.
5. This reservation is now enforced on the Azure disk and the disk will no longer accept writes from VM1. It will only accept writes from VM2.
6. The clustered application can complete the database failover and serve requests from VM2.

The following diagram illustrates another common clustered workload consisting of multiple nodes reading data from the disk for running parallel processes, such as training of machine learning models.



The flow is as follows:

1. The clustered application running on all VMs registers the intent to read or write to the disk.
2. The application instance on VM1 takes an exclusive reservation to write to the disk while opening up reads to the disk from other VMs.
3. This reservation is enforced on your Azure disk.
4. All nodes in the cluster can now read from the disk. Only one node writes back results to the disk, on behalf of all nodes in the cluster.

### Ultra disks reservation flow

Ultra disks offer two extra throttles, for a total of four throttles. Due to this, ultra disks reservation flow can work as described in the earlier section, or it can throttle and distribute performance more granularly.

Reservation Type	Reservation Holder		Registered		Others	
	ReadOnly	ReadWrite	ReadOnly	ReadWrite	ReadOnly	ReadWrite
No Reservation	N/A	N/A	N/A	Yes	N/A	Yes
Write Exclusive	N/A	Yes	Yes	N/A	Yes	N/A
Exclusive Access	N/A	Yes	N/A	N/A	N/A	N/A
Write Exclusive – Registrants Only	N/A	Yes	N/A	Yes	Yes	N/A
Exclusive Access – Registrants Only	N/A	Yes	N/A	Yes	N/A	N/A
Write Exclusive – All Registrants	N/A	Yes	N/A	Yes	Yes	N/A
Exclusive Access – All Registrants	N/A	Yes	N/A	Yes	N/A	N/A

## Performance throttles

### Premium SSD performance throttles

With premium SSD, the disk IOPS and throughput is fixed, for example, IOPS of a P30 is 5000. This value remains whether the disk is shared across 2 VMs or 5 VMs. The disk limits can be reached from a single VM or divided across two or more VMs.

## Ultra disk performance throttles

Ultra disks have the unique capability of allowing you to set your performance by exposing modifiable attributes and allowing you to modify them. By default, there are only two modifiable attributes but, shared ultra disks have two more attributes.

ATTRIBUTE	DESCRIPTION
DiskIOPSReadWrite	The total number of IOPS allowed across all VMs mounting the shared disk with write access.
DiskMBpsReadWrite	The total throughput (MB/s) allowed across all VMs mounting the shared disk with write access.
DiskIOPSReadOnly*	The total number of IOPS allowed across all VMs mounting the shared disk as <code>ReadOnly</code> .
DiskMBpsReadOnly*	The total throughput (MB/s) allowed across all VMs mounting the shared disk as <code>ReadOnly</code> .

\* Applies to shared ultra disks only

The following formulas explain how the performance attributes can be set, since they're user modifiable:

- DiskIOPSReadWrite/DiskIOPSReadOnly:
  - IOPS limits of 300 IOPS/GiB, up to a maximum of 160 K IOPS per disk
  - Minimum of 100 IOPS
  - DiskIOPSReadWrite + DiskIOPSReadOnly is at least 2 IOPS/GiB
- DiskMBpsReadWrite/DiskMBpsReadOnly:
  - The throughput limit of a single disk is 256 KiB/s for each provisioned IOPS, up to a maximum of 2000 MBps per disk
  - The minimum guaranteed throughput per disk is 4KiB/s for each provisioned IOPS, with an overall baseline minimum of 1 MBps

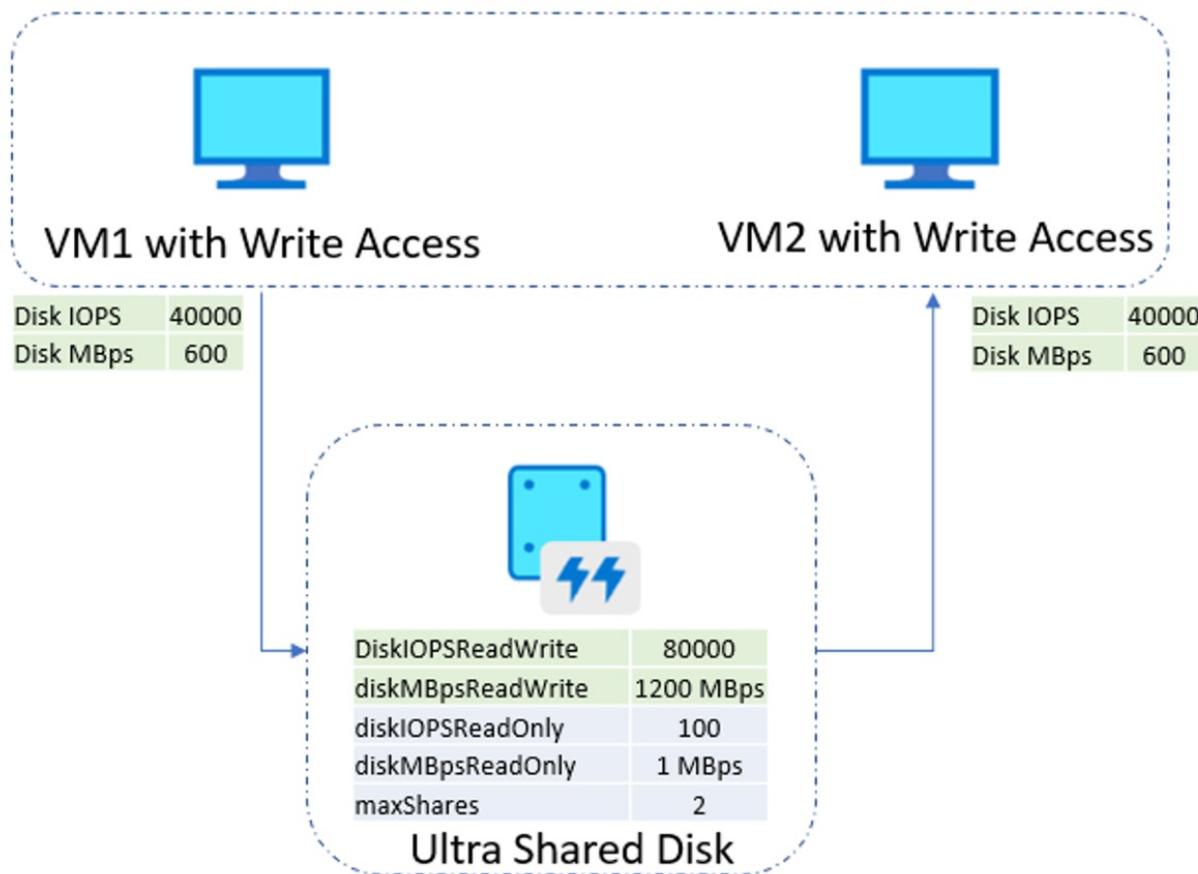
### Examples

The following examples depict a few scenarios that show how the throttling can work with shared ultra disks, specifically.

#### Two nodes cluster using cluster shared volumes

The following is an example of a 2-node WSFC using clustered shared volumes. With this configuration, both VMs have simultaneous write-access to the disk, which results in the `ReadWrite` throttle being split across the two VMs and the `ReadOnly` throttle not being used.

## 2 nodes Windows Cluster using CSV



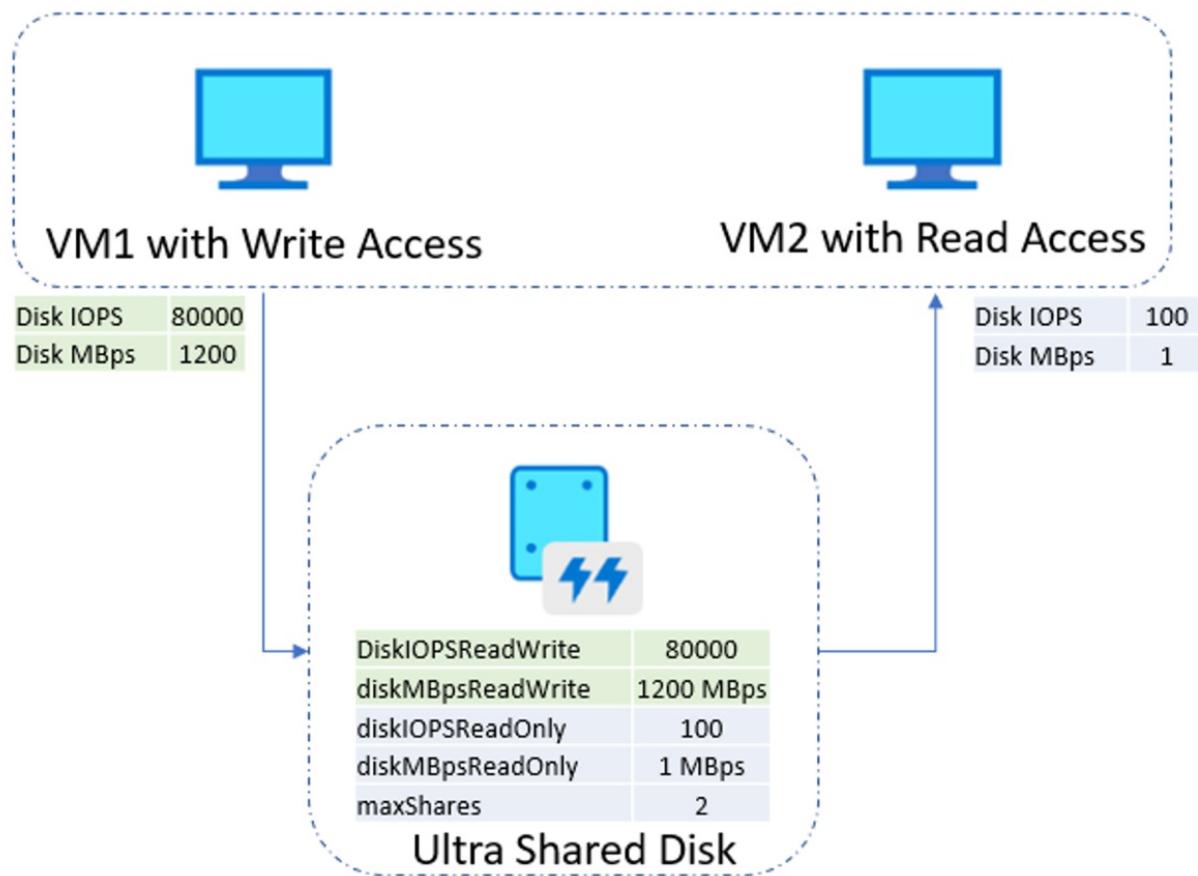
ReadWrite throttles equally split across VMs with write access (2 VMs in this case)

ReadOnly throttles equally split across VMs without write access (None in this case)

### Two node cluster without cluster share volumes

The following is an example of a 2-node WSFC that isn't using clustered shared volumes. With this configuration, only one VM has write-access to the disk. This results in the `ReadWrite` throttle being used exclusively for the primary VM and the `ReadOnly` throttle only being used by the secondary.

## 2 nodes Windows Cluster for Failover not using CSV



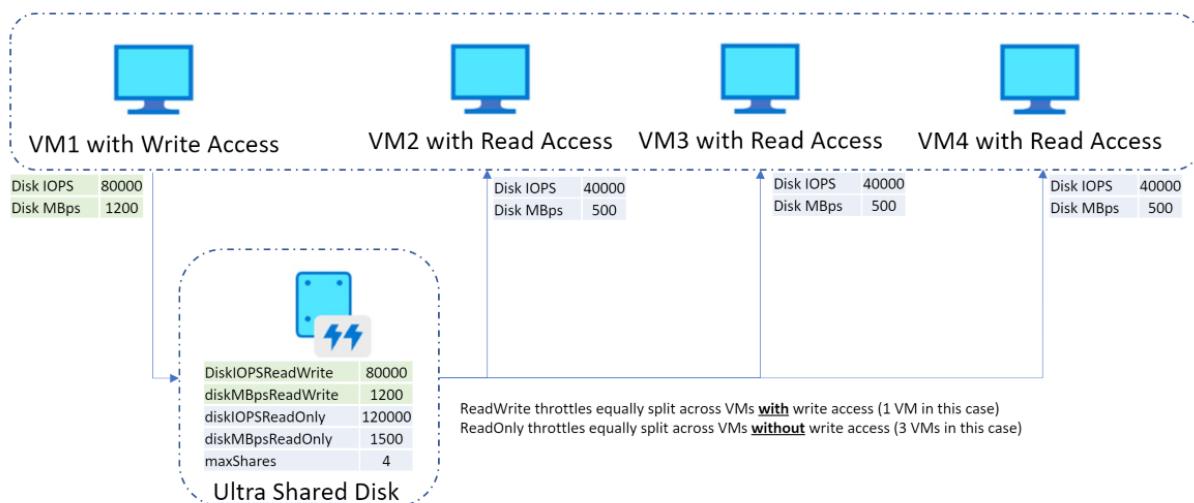
ReadWrite throttles equally split across VMs with write access (1 VM in this case)

ReadOnly throttles equally split across VMs without write access (1 VM in this case)

### Four node Linux cluster

The following is an example of a 4-node Linux cluster with a single writer and three scale-out readers. With this configuration, only one VM has write-access to the disk. This results in the `ReadWrite` throttle being used exclusively for the primary VM and the `ReadOnly` throttle being split by the secondary VMs.

4 nodes Linux cluster with a single Writer and 3 scale out Readers



4 nodes Linux cluster with a single Writer and 3 scale out Readers. ReadWrite throttle will be used for the primary VM while the ReadOnly throttle will be split across the secondary VMs

### Ultra pricing

Ultra shared disks are priced based on provisioned capacity, total provisioned IOPS (diskIOPSReadWrite +

`diskIOPSReadOnly`) and total provisioned Throughput MBps (`diskMBpsReadWrite + diskMBpsReadOnly`). There's no extra charge for each additional VM mount. For example, an ultra shared disk with the following configuration (`diskSizeGB: 1024, DiskIOPSReadWrite: 10000, DiskMBpsReadWrite: 600, DiskIOPSReadOnly: 100, DiskMBpsReadOnly: 1`) is charged with 1024 GiB, 10100 IOPS, and 601 MBps regardless of whether it is mounted to two VMs or five VMs.

## Next steps

If you're interested in enabling and using shared disks for your managed disks, proceed to our article [Enable shared disk](#)

# Enable shared disk

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article covers how to enable the shared disks feature for Azure managed disks. Azure shared disks is a new feature for Azure managed disks that enables you to attach a managed disk to multiple virtual machines (VMs) simultaneously. Attaching a managed disk to multiple VMs allows you to either deploy new or migrate existing clustered applications to Azure.

If you are looking for conceptual information on managed disks that have shared disks enabled, see [Azure shared disks](#).

## Prerequisites

The scripts and commands in this article require either:

- Version 6.0.0 or newer of the Azure PowerShell module.

Or

- The latest version of the Azure CLI.

## Limitations

### General limitations

Enabling shared disks is only available to a subset of disk types. Currently only ultra disks, premium SSD v2 (preview), premium SSDs, and standard SSDs can enable shared disks. Shared disks can be attached to individual VMSS instances but can't be defined in the VMSS models or automatically deployed.

Each managed disk that has shared disks enabled are also subject to the following limitations, organized by disk type:

### Ultra disks

Ultra disks have their own separate list of limitations, unrelated to shared disks. For ultra disk limitations, refer to [Using Azure ultra disks](#).

When sharing ultra disks, they have the following additional limitations:

- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Can't be shared across availability zones.

### Premium SSD v2 (preview)

Premium SSD v2 disks have their own separate list of limitations, unrelated to shared disks. For these limitations, see [Premium SSD v2 limitations](#).

When sharing Premium SSD v2 disks, they have the following additional limitation:

- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Can't be shared across availability zones.

### Premium SSD

- Can only be enabled on data disks, not OS disks.
- Host caching isn't available for premium SSD disks with `maxShares>1`.
- Disk bursting isn't available for premium SSD disks with `maxShares>1`.
- When using Availability sets or virtual machine scale sets with Azure shared disks, [storage fault domain alignment](#) with virtual machine fault domain isn't enforced for the shared data disk.
- When using [proximity placement groups \(PPG\)](#), all virtual machines sharing a disk must be part of the same PPG.
- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Azure Site Recovery support isn't yet available.
- Azure Backup is available through [Azure Disk Backup](#).
- Only [server-side encryption](#) is supported, [Azure Disk Encryption](#) isn't currently supported.
- Can only be shared across availability zones if using [Zone-redundant storage for managed disks](#).

## Standard SSDs

- Can only be enabled on data disks, not OS disks.
- Host caching isn't available for standard SSDs with `maxShares>1`.
- When using Availability sets and virtual machine scale sets with Azure shared disks, [storage fault domain alignment](#) with virtual machine fault domain isn't enforced for the shared data disk.
- When using [proximity placement groups \(PPG\)](#), all virtual machines sharing a disk must be part of the same PPG.
- Only basic disks can be used with some versions of Windows Server Failover Cluster, for details see [Failover clustering hardware requirements and storage options](#).
- Azure Site Recovery support isn't yet available.
- Azure Backup is available through [Azure Disk Backup](#).
- Only [server-side encryption](#) is supported, [Azure Disk Encryption](#) isn't currently supported.
- Can only be shared across availability zones if using [Zone-redundant storage for managed disks](#).

## Supported operating systems

Shared disks support several operating systems. See the [Windows](#) and [Linux](#) sections of the conceptual article for the supported operating systems.

## Disk sizes

For now, only ultra disks, premium SSD v2 (preview), premium SSD, and standard SSDs can enable shared disks. Different disk sizes may have a different `maxShares` limit, which you can't exceed when setting the `maxShares` value.

For each disk, you can define a `maxShares` value that represents the maximum number of nodes that can simultaneously share the disk. For example, if you plan to set up a 2-node failover cluster, you would set `maxShares=2`. The maximum value is an upper bound. Nodes can join or leave the cluster (mount or unmount the disk) as long as the number of nodes is lower than the specified `maxShares` value.

### NOTE

The `maxShares` value can only be set or edited when the disk is detached from all nodes.

## Premium SSD ranges

The following table illustrates the allowed maximum values for `maxShares` by premium SSD sizes:

DISK SIZES	MAXSHARES LIMIT
P1,P2,P3,P4,P6,P10,P15,P20	3
P30, P40, P50	5
P60, P70, P80	10

The IOPS and bandwidth limits for a disk aren't affected by the `maxShares` value. For example, the max IOPS of a P15 disk is 1100 whether `maxShares` = 1 or `maxShares` > 1.

### Standard SSD ranges

The following table illustrates the allowed maximum values for `maxShares` by standard SSD sizes:

DISK SIZES	MAXSHARES LIMIT
E1,E2,E3,E4,E6,E10,E15,E20	3
E30, E40, E50	5
E60, E70, E80	10

The IOPS and bandwidth limits for a disk aren't affected by the `maxShares` value. For example, the max IOPS of a E15 disk is 500 whether `maxShares` = 1 or `maxShares` > 1.

### Ultra disk ranges

The minimum `maxShares` value is 1, while the maximum `maxShares` value is 15. There are no size restrictions on ultra disks, any size ultra disk can use any value for `maxShares`, up to and including the maximum value.

### Premium SSD v2 ranges

The minimum `maxShares` value is 1, while the maximum `maxShares` value is 15. There are no size restrictions on Premium SSD v2, any size Premium SSD v2 disk can use any value for `maxShares`, up to and including the maximum value.

## Deploy shared disks

### Deploy a premium SSD as a shared disk

To deploy a managed disk with the shared disk feature enabled, use the new property `maxShares` and define a value greater than 1. This makes the disk shareable across multiple VMs.

#### IMPORTANT

The value of `maxShares` can only be set or changed when a disk is unmounted from all VMs. See the [Disk sizes](#) for the allowed values for `maxShares`.

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)
- [Resource Manager Template](#)

1. Sign in to the Azure portal.

2. Search for and Select Disks.
3. Select + Create to create a new managed disk.
4. Fill in the details and select an appropriate region, then select Change size.

The screenshot shows the 'Create a managed disk' wizard in the Microsoft Azure portal. The 'Basics' tab is selected. In the 'Disk details' section, the 'Size' dropdown is set to '16 GiB' and 'Premium SSD LRS'. A red box highlights the 'Change size' link below the dropdown. The left sidebar shows various Azure services like App Services, SQL databases, and Virtual machines.

5. Select the premium SSD size and SKU that you want and select OK.

Locally-redundant storage (data is replicated within a single datacenter)	
<b>Premium SSD</b>	Best for production and performance sensitive workloads
<b>Standard SSD</b>	Best for web servers, lightly used enterprise applications and dev/test
<b>Standard HDD</b>	Best for backup, non-critical, and infrequent access

Zone-redundant storage (data is replicated within multiple datacenters)	
<b>Premium SSD</b>	Best for the production workloads that need storage resiliency against zone failures
<b>Standard SSD</b>	Best for web servers, lightly used enterprise applications and dev/test that need storage resiliency against zone failures

6. Proceed through the deployment until you get to the Advanced pane.
7. Select Yes for Enable shared disk and select the amount of Max shares you want.

**Shared disk**

Allow this disk to be attached to two or more virtual machines, depending on storage type and disk size. When shared disk is enabled host caching is unavailable. [Learn more about shared disks](#)

Yes  No

Max shares ⓘ

8. Select **Review + Create**.

### Deploy a standard SSD as a shared disk

To deploy a managed disk with the shared disk feature enabled, use the new property `maxShares` and define a value greater than 1. This makes the disk shareable across multiple VMs.

#### IMPORTANT

The value of `maxShares` can only be set or changed when a disk is unmounted from all VMs. See the [Disk sizes](#) for the allowed values for `maxShares`.

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)
- [Resource Manager Template](#)

1. Sign in to the Azure portal.
2. Search for and **Select Disks**.
3. Select **+ Create** to create a new managed disk.
4. Fill in the details and select an appropriate region, then select **Change size**.

The screenshot shows the 'Create a managed disk' wizard in the Microsoft Azure portal. The 'Basics' tab is selected. In the 'Disk details' section, the 'Size' field is highlighted with a red box, showing '16 GiB Premium SSD LRS' and a 'Change size' link. Other fields include 'Subscription' (My Example Subscription), 'Resource group' (Create new), 'Disk name' (empty), 'Region' (US West US), 'Availability zone' (None), and 'Source type' (None). At the bottom, there are 'Review + create' and 'Next : Encryption >' buttons.

5. Select the standard SSD size and SKU that you want and select **OK**.

<b>Locally-redundant storage (data is replicated within a single datacenter)</b>
Premium SSD
Best for production and performance sensitive workloads
<b>Standard SSD</b>
Best for web servers, lightly used enterprise applications and dev/test
Standard HDD
Best for backup, non-critical, and infrequent access
<b>Zone-redundant storage (data is replicated within multiple datacenters)</b>
Premium SSD
Best for the production workloads that need storage resiliency against zone failures
<b>Standard SSD</b>
Best for web servers, lightly used enterprise applications and dev/test that need storage resiliency against zone failures

6. Proceed through the deployment until you get to the **Advanced** pane.

7. Select **Yes** for **Enable shared disk** and select the amount of **Max shares** you want.

<b>Shared disk</b>
Allow this disk to be attached to two or more virtual machines, depending on storage type and disk size. When shared disk is enabled host caching is unavailable. <a href="#">Learn more about shared disks</a>
<input checked="" type="radio"/> Yes <input type="radio"/> No
Max shares <a href="#">(i)</a> <input type="text" value="2"/> <a href="#">▼</a>

8. Select **Review + Create**.

### Deploy an ultra disk as a shared disk

To deploy a managed disk with the shared disk feature enabled, change the `maxShares` parameter to a value greater than 1. This makes the disk shareable across multiple VMs.

#### IMPORTANT

The value of `maxShares` can only be set or changed when a disk is unmounted from all VMs. See the [Disk sizes](#) for the allowed values for `maxShares`.

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)
- [Resource Manager Template](#)

1. Sign in to the Azure portal.
2. Search for and Select Disks.
3. Select **+ Create** to create a new managed disk.

4. Fill in the details, then select **Change size**.

5. Select ultra disk for the **Disk SKU**.

The screenshot shows a dropdown menu titled "Disk SKU ⓘ". The "Ultra Disk (locally-redundant storage)" option is selected, indicated by a blue border. Below it, other options are listed: "Locally-redundant storage (data is replicated within a single datacenter)", "Premium SSD", "Standard SSD", "Standard HDD", and "Ultra Disk". The "Ultra Disk" option is highlighted with a grey background. Descriptions for each option are provided below them.

Disk SKU	Description
Ultra Disk (locally-redundant storage)	Locally-redundant storage (data is replicated within a single datacenter) Best for production and performance sensitive workloads
Premium SSD	Standard SSD Best for web servers, lightly used enterprise applications and dev/test
Standard SSD	Standard HDD Best for backup, non-critical, and infrequent access
Standard HDD	Ultra Disk Best for IO-intensive workloads such as SAP HANA, top tier databases (for example, SQL, Oracle), and other transaction-heavy workloads

6. Select the disk size that you want and select **OK**.

7. Proceed through the deployment until you get to the **Advanced** pane.

8. Select **Yes** for **Enable shared disk** and select the amount of **Max shares** you want.

9. Select **Review + Create**.

Add additional configurations for your managed disk

#### Shared disk

Allow this disk to be attached to two or more virtual machines, depending on storage type and disk size. When shared disk is enabled host caching is unavailable. [Learn more about shared disks](#)

Enable shared disk

Yes    No

Max shares ⓘ

▼

#### Ultra disk

Read/write disk IOPS \* ⓘ

Enter a number between 100 and 4800

Read/write disk throughput (MB/s) \* ⓘ

Enter a number between 1 and 30

Read-only disk IOPS \* ⓘ

Enter a number between 100 and 4800

Read-only disk throughput (MB/s) \* ⓘ

Enter a number between 1 and 25

#### Ultra disk

Logical sector size (bytes) ⓘ

4096    512

**Review + create**

< Previous

Next : Tags >

## Share an existing disk

To share an existing disk, or update how many VMs it can mount to, set the `maxShares` parameter with either the Azure PowerShell module or Azure CLI. You can also set `maxShares` to 1, if you want to disable sharing.

#### IMPORTANT

The value of `maxShares` can only be set or changed when a disk is unmounted from all VMs. See the [Disk sizes](#) for the allowed values for `maxShares`. Before detaching a disk, record the LUN ID for when you re-attach it.

## PowerShell

```
$datadiskconfig = Get-AzDisk -DiskName "mySharedDisk"
$datadiskconfig.maxShares = 3

Update-AzDisk -ResourceGroupName 'myResourceGroup' -DiskName 'mySharedDisk' -Disk $datadiskconfig
```

## CLI

```
#Modifying a disk to enable or modify sharing configuration

az disk update --name mySharedDisk --max-shares 5
```

# Using Azure shared disks with your VMs

Once you've deployed a shared disk with `maxShares>1`, you can mount the disk to one or more of your VMs.

## NOTE

If you are deploying an ultra disk, make sure it matches the necessary requirements. See [Using Azure ultra disks](#) for details.

```
$resourceGroup = "myResourceGroup"
$location = "WestCentralUS"

$vm = New-AzVm -ResourceGroupName $resourceGroup -Name "myVM" -Location $location -VirtualNetworkName
"myVnet" -SubnetName "mySubnet" -SecurityGroupName "myNetworkSecurityGroup" -PublicIpAddressName
"myPublicIpAddress"

$dataDisk = Get-AzDisk -ResourceGroupName $resourceGroup -DiskName "mySharedDisk"

$vm = Add-AzVMDataDisk -VM $vm -Name "mySharedDisk" -CreateOption Attach -ManagedDiskId $dataDisk.Id -Lun 0

update-AzVm -VM $vm -ResourceGroupName $resourceGroup
```

## Supported SCSI PR commands

Once you've mounted the shared disk to your VMs in your cluster, you can establish quorum and read/write to the disk using SCSI PR. The following PR commands are available when using Azure shared disks:

To interact with the disk, start with the persistent-reservation-action list:

```
PR_REGISTER_KEY
PR_REGISTER_AND_IGNORE
PR_GET_CONFIGURATION
PR_RESERVE
PR_PREEMPT_RESERVATION
PR_CLEAR_RESERVATION
PR_RELEASE_RESERVATION
```

When using PR\_RESERVE, PR\_PREEMPT\_RESERVATION, or PR\_RELEASE\_RESERVATION, provide one of the following persistent-reservation-type:

```
PR_NONE  
  
PR_WRITE_EXCLUSIVE  
  
PR_EXCLUSIVE_ACCESS  
  
PR_WRITE_EXCLUSIVE_REGISTRANTS_ONLY  
  
PR_EXCLUSIVE_ACCESS_REGISTRANTS_ONLY  
  
PR_WRITE_EXCLUSIVE_ALL_REGISTRANTS  
  
PR_EXCLUSIVE_ACCESS_ALL_REGISTRANTS
```

You also need to provide a persistent-reservation-key when using PR\_RESERVE, PR\_REGISTER\_AND\_IGNORE, PR\_REGISTER\_KEY, PR\_PREEMPT\_RESERVATION, PR\_CLEAR\_RESERVATION, or PR\_RELEASE\_RESERVATION.

## Next steps

If you prefer to use Azure Resource Manager templates to deploy your disk, the following sample templates are available:

- [Premium SSD](#)
- [Regional ultra disks](#)
- [Zonal ultra disks](#)

# Azure disk pools (preview)

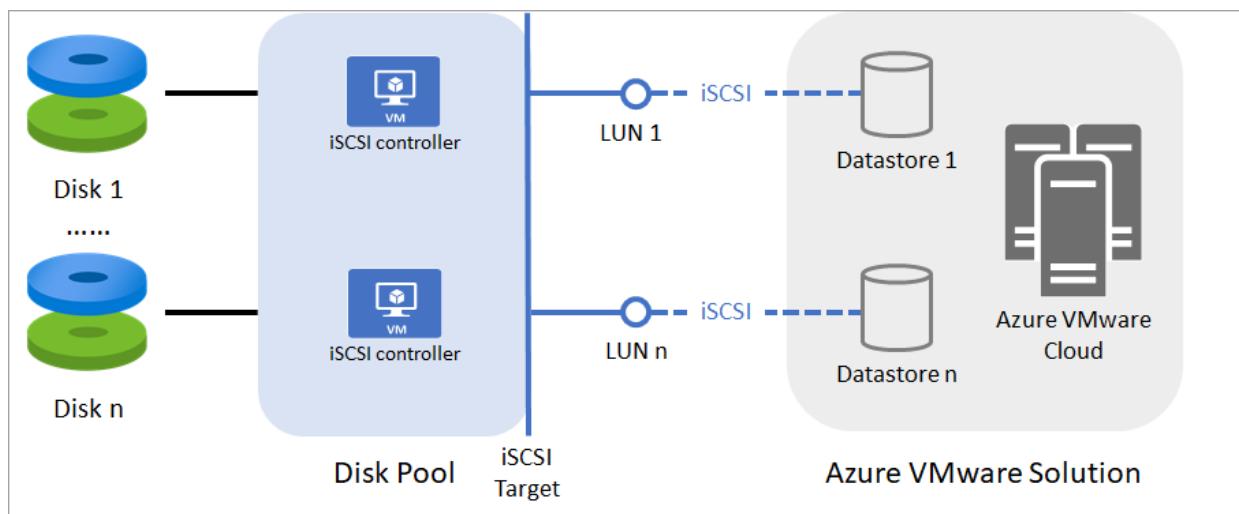
9/21/2022 • 2 minutes to read • [Edit Online](#)

An Azure disk pool (preview) is an Azure resource that allows your applications and workloads to access a group of managed disks from a single endpoint. A disk pool can expose an Internet Small Computer Systems Interface (iSCSI) target to enable data access to disks inside this pool over iSCSI. Each disk pool can have one iSCSI target and each disk can be exposed as an iSCSI LUN. You can connect disks under the disk pool to Azure VMware Solution hosts as datastores. This allows you to scale your storage independent of your Azure VMware Solution hosts. Once a datastore is configured, you can create volumes on it and attach them to your VMware instances.

## How it works

When a disk pool is deployed, a managed resource group is automatically created for you. This managed resource group contains all Azure resources necessary for the operation of a disk pool. The naming convention for these resource groups is: MSP\_(resource-group-name)\_(diskpool-name)\_(region-name).

When you add a managed disk to the disk pool, the disk is attached to managed iSCSI controllers. Multiple managed disks can be added as storage targets to a disk pool, each storage target is presented as an iSCSI LUN under the disk pool's iSCSI target. Disk pools offer native support for Azure VMware Solution. An Azure VMware Solution cluster can connect to a disk pool, which would encompass all Azure VMware Solution hosts in that environment. The following diagram shows how you can use disk pools with Azure VMware Solution.



## Restrictions

In preview, disk pools have the following restrictions:

- Only premium SSDs and standard SSDs, or ultra disks can be added to a disk pool.
  - A disk pool can't be configured to contain both ultra disks and premium/standard SSDs. If a disk pool is configured to use ultra disks, it can only contain ultra disks. Likewise, a disk pool configured to use premium and standard SSDs can only contain premium and standard SSDs.
- Disks using [zone-redundant storage \(ZRS\)](#) aren't currently supported.

## Regional availability

Disk pools are currently available in the following regions:

- Australia East

- Canada Central
- Central US
- East US
- East US 2
- West US 2
- Japan East
- North Europe
- West Europe
- Southeast Asia
- UK South
- Korea Central
- Sweden Central
- Central India

## Billing

When you deploy a disk pool, there are two areas that will incur billing costs: The price of the disk pool service fee itself, and the price of each individual disk added to the pool. For example, if you have a disk pool with one P30 disk added, you will be billed for the P30 disk and the disk pool. Other than the disk pool and your disks, there are no extra service charges for a disk pool and you will not be billed for the resources deployed in the managed resource group: `MSP_(resource-group-name)(diskpool-name)(region-name)`.

See the [Azure managed disk pricing page](#) for regional pricing on disk pools and disks to evaluate the cost of a disk pool for you.

## Next steps

See the [disk pools planning guide](#).

# Azure disk pools (preview) planning guide

9/21/2022 • 4 minutes to read • [Edit Online](#)

It's important to understand the performance requirements of your workload before you deploy an Azure disk pool (preview). Determining your requirements in advance allows you to get the most performance out of your disk pool. The performance of a disk pool is determined by three main factors, adjusting any of them will tweak your disk pool's performance:

- The disk pool's scalability target
- The scalability targets of individual disks contained in the disk pool
- The networking connection between the client machines to the disk pool.

## Optimize for low latency

If you're prioritizing for low latency, add ultra disks to your disk pool. Ultra disks provide sub-millisecond disk latency. To get the lowest latency possible, you must also evaluate your network configuration and ensure it's using the most optimal path. Consider using [ExpressRoute FastPath](#) to minimize network latency.

## Optimize for high throughput

If you're prioritizing throughput, begin by evaluating the performance targets of the different disk pool SKUs, as well as the number of disk pools required to deliver your throughput targets. If your performance needs exceed what a premium disk pool can provide, you can split your deployment across multiple disk pools. Then, you can decide how to best utilize the performance offered on a disk pool amongst each individual disk and their types. For a disk pool, you can either mix and match between premium and standard SSDs, or use ultra disks only. Ultra disks can't be used with premium or standard SSDs. Select the disk type that best fits your needs. Also, confirm the network connectivity from your clients to the disk pool is not a bottleneck, especially the throughput.

## Use cases

The following table lists some typical use cases for disk pools with Azure VMware Solution and a recommended configuration.

AZURE VMWARE SOLUTION USE CASES	SUGGESTED DISK TYPE	SUGGESTED DISK POOL SKU	SUGGESTED NETWORK CONFIGURATION
Block storage for active working sets, like an extension of Azure VMware Solution vSAN.	Ultra disks	Premium	Use ExpressRoute virtual network gateway: Ultra Performance or ErGw3AZ (10 Gbps) to connect the disk pool virtual network to the Azure VMware Solution cloud and enable FastPath to minimize network latency.

AZURE VMWARE SOLUTION USE CASES	SUGGESTED DISK TYPE	SUGGESTED DISK POOL SKU	SUGGESTED NETWORK CONFIGURATION
Tiering - tier infrequently accessed data from the Azure VMware Solution vSAN to the disk pool.	Premium SSD, standard SSD	Standard	Use ExpressRoute virtual network gateway: Standard (1 Gbps) or High Performance (2 Gbps) to connect the disk pool virtual network to the Azure VMware Solution cloud.
Data storage for disaster recovery site on Azure VMware Solution: replicate data from on-premises or primary VMware environment to the disk pool as a secondary site.	Premium SSD, standard SSD	Standard, Basic	Use ExpressRoute virtual network gateway: Standard (1 Gbps) or High Performance (2 Gbps) to connect the disk pool virtual network to the Azure VMware Solution cloud.

Refer to the [Networking planning checklist for Azure VMware Solution](#) to plan for your networking setup, along with other Azure VMware Solution considerations.

## Disk pool scalability and performance targets

RESOURCE	BASIC DISK POOL	STANDARD DISK POOL	PREMIUM DISK POOL
Maximum number of disks per disk pool	16	32	32
Maximum IOPS per disk pool	12,800	25,600	51,200
Maximum MBps per disk pool	192	384	768

The following example should give you an idea of how the different performance factors work together:

As an example, if you add two 1-TiB premium SSDs (P30, with a provisioned target of 5000 IOPS and 200 Mbps) into a standard disk pool, you can achieve  $2 \times 5000 = 10,000$  IOPS. However, throughput would be capped at 384 MBps by the disk pool. To exceed this 384-MBps limit, you can deploy more disk pools to scale out for extra throughput. Your network throughput will limit the effectiveness of scaling out.

Disk pools created without specifying the SKU in the REST API are the standard disk pool, by default.

## Availability

Disk pools are currently in preview, and shouldn't be used for production workloads. By default, a disk pool only supports premium and standard SSDs. You can enable support for ultra disks on a disk pool instead but, a disk pool with ultra disks isn't compatible with premium or standard SSDs.

Disk pools with support for premium and standard SSDs are based on a highly available architecture, with multiples hosting the iSCSI endpoint. Disk pools with support for ultra disks are hosted on a single instance deployment.

If your disk pool becomes inaccessible to your Azure VMware Solution cloud for any reason, you will experience the following:

- All datastores associated to the disk pool will no longer be accessible.
- All VMware VMs hosted in the Azure VMware Solution cloud that is using the impacted datastores will be in an unhealthy state.
- The health of clusters in the Azure VMware Solution cloud won't be impacted, except for one operation: You won't be able to place a host into maintenance mode. Azure VMware Solution will handle this failure and attempt recovery by disconnecting the impacted datastores.

## Next steps

- [Deploy a disk pool](#).
- To learn about how Azure VMware Solutions integrates disk pools, see [Attach disk pools to Azure VMware Solution hosts \(Preview\)](#).

# Deploy an Azure disk pool (preview)

9/21/2022 • 8 minutes to read • [Edit Online](#)

This article covers how to deploy and configure an Azure disk pool (preview). Before deploying a disk pool, read the [conceptual](#) and [planning](#) articles.

For a disk pool to work correctly, you must complete the following steps:

- Register your subscription for the preview.
- Delegate a subnet to your disk pool.
- Assign the resource provider of disk pool role-based access control (RBAC) permissions for managing your disk resources.
- Create the disk pool.
  - Add disks to your disk pool.

## Prerequisites

To successfully deploy a disk pool, you must have:

- A set of managed disks you want to add to a disk pool.
  - A virtual network with a dedicated subnet deployed for your disk pool.
    - Outbound ports 53, 443, and 5671 must be open.
    - Ensure that your network setting don't block any of your disk pool's required outbound dependencies.
- You can use either the [Azure PowerShell module](#) or [Azure CLI](#) to get the complete list of all outbound dependencies.

If you're going to use the Azure PowerShell module, install [version 6.1.0 or newer](#).

If you're going to use the Azure CLI, install [the latest version](#).

## Register your subscription for the preview

Register your subscription to the **Microsoft.StoragePool** provider, to be able to create and use disk pools.

1. Sign in to the [Azure portal](#).
2. On the Azure portal menu, search for and select **Subscriptions**.
3. Select the subscription you want to use for disk pools.
4. On the left menu, under **Settings**, select **Resource providers**.
5. Find the resource provider **Microsoft.StoragePool** and select **Register**.

Once your subscription has been registered, you can deploy a disk pool.

## Delegate subnet permission

For your disk pool to work with your client machines, you must delegate a subnet to your Azure disk pool. When creating a disk pool, you specify a virtual network and the delegated subnet. You may either create a new subnet or use an existing one and delegate to the **Microsoft.StoragePool/diskPools** resource provider.

1. Go to the virtual networks pane in the Azure portal and select the virtual network to use for the disk pool.
2. Select **Subnets** from the virtual network pane and select **+ Subnet**.
3. Create a new subnet by completing the following required fields in the **Add subnet** pane: - Subnet

delegation: Select Microsoft.StoragePool/diskPools

For more information on subnet delegation, see [Add or remove a subnet delegation](#).

## Assign StoragePool resource provider permissions

For a disk to be able to be used in a disk pool, it must meet the following requirements:

- The **StoragePool** resource provider must have been assigned an RBAC role that contains **Read** and **Write** permissions for every managed disk in the disk pool.
- Must be either a premium SSD, standard SSD, or an ultra disk in the same availability zone as the disk pool.
  - For ultra disks, it must have a disk sector size of 512 bytes.
- Disk pools can't be configured to contain both premium/standard SSDs and ultra disks. A disk pool configured for ultra disks can only contain ultra disks. Likewise, a disk pool configured for premium or standard SSDs can only contain premium and standard SSDs.
- Must be a shared disk with a maxShares value of two or greater.

1. Sign in to the [Azure portal](#).
2. Search for and select either the resource group that contains the disks or each disk themselves.
3. Select **Access control (IAM)**.
4. Select **Add role assignment (Preview)**, and select **Disk Pool Operator** in the role list.

If you prefer, you may create your own custom role instead. A custom role for disk pools must have the following RBAC permissions to function: **Microsoft.Compute/disks/write** and **Microsoft.Compute/disks/read**.

5. For **Assign access to**, select **User, group, or service principal**.
6. Select **+ Select members** and then search for **StoragePool Resource Provider**, select it, and save.

## Create a disk pool

For optimal performance, deploy the disk pool in the same Availability Zone of your clients. If you are deploying a disk pool for an Azure VMware Solution cloud and need guidance on identifying the Availability Zone, fill in this [form](#).

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Search for and select **Disk pool**.
2. Select **+ Add** to create a new disk pool.
3. Fill in the details requested, select the same region and availability zone as the clients that will use the disk pool.
4. Select the subnet that has been delegated to the **StoragePool** resource provider, and its associated virtual network.
5. Select **Next** to add disks to your disk pool.

The screenshot shows the Microsoft Azure portal interface for creating a disk pool. The left sidebar includes links for 'Create a resource', 'Home', 'Dashboard', 'All services', 'FAVORITES' (with items like All resources, Resource groups, Cognitive Search, App Services, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, Monitor, Advisor, Security Center, Help + support, and Cost Management + Billing), and a search bar at the top.

The main content area is titled 'Create a disk pool'. The 'Basics' tab is active. A sub-instruction says 'Create a disk pool with iSCSI support and scale block storage independent of compute.' with a 'Learn more' link. The 'Project details' section asks to select a subscription to manage deployed resources and costs, using resource groups to organize and manage all your resources. It shows 'Subscription \*' set to 'My Example Subscription' and 'Resource group \*' as an empty dropdown with a 'Create new' button.

The 'Instance details' section contains fields for 'Name \*' (empty input), 'Region \*' (set to '(US) West US 2'), 'Availability zone \*' (set to '1'), 'Virtual network \*' (empty dropdown), and 'Disk pool SKU \*' (empty dropdown). At the bottom are buttons for 'Review + create' (highlighted in blue), '< Previous', and 'Next : Disks >'.

## Add disks

### Prerequisites

To add a disk, it must meet the following requirements:

- Must be either a premium SSD, standard SSD, or an ultra disk in the same availability zone as the disk pool.
  - Currently, you can only add premium SSDs and Standard SSDs in the portal. Ultra disks must be added with either the Azure PowerShell module or the Azure CLI.
  - For ultra disks, it must have a disk sector size of 512 bytes.
- Must be a shared disk with a maxShares value of two or greater.
- Disk pools can't be configured to contain both premium/standard SSDs and ultra disks. A disk pool configured for ultra disks can only contain ultra disks. Likewise, a disk pool configured for premium or standard SSDs can only contain premium and standard SSDs.
- You must grant RBAC permissions to the resource provider of disk pool to manage the disk you plan to add.

If your disk meets these requirements, you can add it to a disk pool by selecting **+ Add disk** in the disk pool pane.

# Create a disk pool

X

Basics   Disks   iSCSI   Tags   Review + create

Select existing Azure Managed Disks to add to the disk pool. Only premium SSD or standard SSD deployed in the same region and availability zone as the disk pool can be added to the pool. [Learn more](#)

 In order to successfully add disks to a disk pool, the following prerequisites are necessary:

- Must be a premium SSD or standard SSD in the same region and availability zone as the disk pool.
- Enabled for shared disks with maxShares value set to two or greater.
- You must grant required RBAC permissions to the disk pool resource provider. [Learn more](#)

 Attach existing disk  Remove disk from disk pool

<input type="checkbox"/> Disk name	Size (GiB)	Disk type	IOPS	Bandwidth (MB)
------------------------------------	------------	-----------	------	----------------

## Enable iSCSI

1. Select the iSCSI pane.
2. Select **Enable iSCSI**.
3. Enter the name of the iSCSI target, the iSCSI target IQN will generate based on this name.
  - The ACL mode is set to **Dynamic** by default. To use your disk pool as a storage solution for Azure VMware Solution, the ACL mode must be set to **Dynamic**.
4. Select **Review + create**.

## Create a disk pool

Basics Disks iSCSI Tags Review + create

Enable iSCSI support on your disk pool and allow client connections over iSCSI protocol. One iSCSI target can be configured and exposed per disk pool. [Learn more](#)

Enable iSCSI [\(i\)](#)

iSCSI target name [\\*](#) [\(i\)](#)

example [\(v\)](#)

 iSCSI target name cannot be changed after iSCSI target is created.

iSCSI target IQN [\(i\)](#)

iqn.2021-10.com.microsoft@example

### Access control list (ACL) options

Access control lists are used to specify the clients (iSCSI initiators) allowed to access the disk pool. With default set to Dynamic, the disk pool will be open to all incoming connections within the vNet. [Learn more](#)

ACL mode [\\*](#)

Dynamic  Static

### Disks enabled for iSCSI

Enable iSCSI connection for disks. To add disks, see the disks tab. [Learn more](#)

Enable iSCSI protocol for all attached disks to this disk pool

[Review + create](#)

[< Previous](#)

[Next : Tags >](#)

## Next steps

- If you encounter any issues deploying a disk pool, see [Troubleshoot Azure disk pools \(preview\)](#).
- [Attach disk pools to Azure VMware Solution hosts \(Preview\)](#).
- [Manage a disk pool](#).

# Move a disk pool (preview) to a different subscription

9/21/2022 • 2 minutes to read • [Edit Online](#)

Moving an Azure disk pool (preview) involves moving the disk pool itself, the disks contained in the disk pool, the disk pool's managed resource group, and all the resources contained in the managed resource group. Currently, Azure doesn't support moving multiple resource groups to another subscription at once.

- Export the template of your existing disk pool.
- Delete the old disk pool.
- Move the Azure resources necessary to create a disk pool.
- Redeploy the disk pool.

## Export your existing disk pool template

To make the redeployment process simpler, export the template from your existing disk pool. You can use this template to redeploy the disk pool in a subscription of your choice, with the same configuration. See [this article](#) to learn how to export a template from a resource.

## Delete the old disk pool

Now that you've exported the template, delete the old disk pool. Deleting the disk pool removes the disk pool resource and its managed resource group. See [this article](#) for guidance on how to delete a disk pool.

## Move your disks and virtual network

Now that the disk pool is deleted, you can move the virtual network and your disks, and potentially your clients, to the subscription you want to change to. See [this article](#) to learn how to move Azure resources to another subscription.

## Redeploy your disk pool

Once you've moved your other resources into the subscription, update the template of your old disk pool so that all the references to your disks, virtual network, subnet, and clients, all now point to their new resource URIs. Once you've done that, redeploy the template to the new subscription. To learn how to edit and deploy a template, see [this article](#).

## Next steps

To learn how to manage your disk pool, see [Manage a disk pool](#).

# Manage an Azure disk pool (preview)

9/21/2022 • 6 minutes to read • [Edit Online](#)

This article covers how to add a managed disk to an Azure disk pool (preview) and how to disable iSCSI support on a disk that has been added to a disk pool.

## Add a disk to a pool

Your disk must meet the following requirements in order to be added to the disk pool:

- Must be either a premium SSD, standard SSD, or an ultra disk in the same region and availability zone as the disk pool.
    - Ultra disks must have a disk sector size of 512 bytes.
  - Must be a shared disk, with a maxShares value of two or greater.
  - You must [provide the StoragePool resource provider RBAC permissions to the disks that will be added to the disk pool](#).
- [Portal](#)
  - [PowerShell](#)
  - [Azure CLI](#)

1. Sign in to the [Azure portal](#).
2. Navigate to your disk pool, and select **Disks** under **Settings**.
3. Select **Attach existing disk** and select your disks.
4. When you have chosen all the disks you'd like to attach, select **Save**.

Disk name	Size (GiB)	Disk type	IOPS	Bandwidth (Mbps)
poolOne	1024	Premium SSD LRS	5000	200
poolThree	1024 GiB	Premium SSD LRS (5000 IOPS, 200 Mbps bandwidth)	-	-
poolTwo	1024 GiB	Premium SSD LRS (5000 IOPS, 200 Mbps bandwidth)	0	0

Now that you've attached your disk, you must enable their LUNS.

5. Select **iSCSI** under **Settings**.
6. Select **Add LUN** under **Disks enabled for iSCSI**.

7. Select the disk you attached earlier.

8. Select **Save**.

The screenshot shows the Azure portal interface for managing a disk pool. The left sidebar lists various tabs: Overview, Activity log, Access control (IAM), Tags, Settings, Disks, iSCSI (which is selected and highlighted in grey), Properties, Locks, Automation, Tasks (preview), Export template, Support + troubleshooting, and New Support Request. The main content area is titled 'yourdiskpool | iSCSI'. It includes sections for enabling iSCSI support, setting the iSCSI target name (example), iSCSI target IQN (iqn.2021-10.com.example:example), iSCSI resource ID, and endpoints (1.1.1.2:1 | 1.1.1.1:1). Below these, the 'Access control list (ACL) options' section shows 'ACL mode' set to 'Dynamic'. The 'Disks enabled for iSCSI' section lists four disks: poolOne, poolThree (selected and highlighted with a red border), and poolTwo. The 'Save' button at the bottom is also highlighted with a red border.

Now that you've attached your disk and enabled the LUN, you must create and attach it as an iSCSI datastore to your Azure VMware Solution private cloud. See [Attach the iSCSI LUN](#) for details.

## Disable iSCSI on a disk and remove it from the pool

Before you disable iSCSI support on a disk, confirm there is no outstanding iSCSI connections to the iSCSI LUN the disk is exposed as. When a disk is removed from the disk pool, it isn't automatically deleted. This prevents any data loss but you will still be billed for storing data. If you don't need the data stored in a disk, you can manually delete the disk. This will delete the disk and all data stored on it and will prevent further charges.

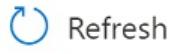
- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Sign in to the [Azure portal](#).
2. Navigate to your disk pool and select iSCSI under **Settings**.
3. Under **Disks enabled for iSCSI** select the disks you'd like to remove and select **Remove LUN**.
4. Select **Save** and wait for the operation to complete.



Disk Pool

&lt;&lt;

[Overview](#)

Enable iSCSI support on your disk pool

[Activity log](#)[Enable iSCSI](#) ⓘ[Access control \(IAM\)](#)

iSCSI target name ⓘ

[Tags](#)

iSCSI target IQN ⓘ

**Settings**

iSCSI resource ID

[Disks](#)

Endpoints ⓘ

[iSCSI](#)**Access control list (ACL) options**[Properties](#)Access control lists are used to specify connections within the vNet. [Learn more](#)[Locks](#)

ACL mode

**Automation****Disks enabled for iSCSI**[Tasks \(preview\)](#)

Enable iSCSI connection for disks. To

[Export template](#)[Add LUN](#)[Remove LUN](#)**Support + troubleshooting**[New Support Request](#) Disk name poolOne poolThree poolTwo

Now that you've disabled the LUN, you can remove your disks from the disk pool.

5. Select **Disks** under **Settings**.
6. Select **Remove disk from disk pool** and select your disks.
7. Select **Save**.

When the operation completes, your disk will have been completely removed from the disk pool.

 Attach existing disk Remove disk from disk pool

---

Disk name

poolOne

poolThree

poolTwo

## Next steps

- To learn how to move a disk pool to another subscription, see [Move a disk pool to a different subscription](#).
- To learn how to deprovision a disk pool, see [Deprovision an Azure disk pool](#).

# Deprovision an Azure disk pool (preview)

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article covers the deletion process for an Azure disk pool (preview) and how to disable iSCSI support.

## Stop a disk pool

You can stop a disk pool to save costs and preserve all configurations. When a disk pool is stopped, you can no longer connect to it over iSCSI. The managed resources deployed to support the disk pool will not be deleted. You must disconnect all clients with iSCSI connections to the disk pool first before stopping a disk pool. You can start a disk pool at any time. This will reactivate the iSCSI target exposed on this disk pool.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Sign in to the [Azure portal](#).
2. Navigate to your disk pool, and select **Stop**.

## Disable iSCSI support

If you disable iSCSI support on a disk pool, you can no longer connect to a disk pool.

When you first enable iSCSI support on a disk pool, an iSCSI target is created as the endpoint for the iSCSI connection. You can disable iSCSI support on the disk pool by deleting the iSCSI target. Each disk pool can only have one iSCSI target configured.

You can re-enable iSCSI support on an existing disk pool. iSCSI support cannot be disabled on the disk pool if there are outstanding iSCSI connections to the disk pool.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Search for **Disk pool** and select your disk pool.
2. Select **iSCSI** under **Settings**.
3. Clear the **Enable iSCSI** checkbox and select **Save**.

## Delete a disk pool

When you delete a disk pool, all the resources in the managed resource group are also deleted. If there are outstanding iSCSI connections to the disk pool, you cannot delete the disk pool. You must disconnect all clients with iSCSI connections to the disk pool first. Disks that have been added to the disk pool will not be deleted.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Sign in to the [Azure portal](#).
2. Search for **Disk pool** and select it, then select the disk pool you want to delete.

3. Select **Delete** at the top of the pane.

## Next steps

Learn about [Azure managed disks](#).

# Troubleshoot Azure disk pools (preview)

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article lists some common failure codes related to Azure disk pools (preview). It also provides possible resolutions and some clarity on disk pool statuses.

## Disk pool status

Disk pools and iSCSI targets each have four states: **Unknown**, **Running**, **Updating**, and **Stopped (deallocated)**.

**Unknown** means that the resource is in a bad or unknown state. To attempt recovery, perform an update operation on the resource (such as adding or removing disks/LUNS) or delete and redeploy your disk pool.

**Running** means the resource is running and in a healthy state.

**Updating** means that the resource is going through an update. This usually happens during deployment or when applying an update like adding disks or LUNs.

**Stopped (deallocated)** means that the resource is stopped and its underlying resources have been deallocated. You can restart the resource to recover your disk pool or iSCSI target.

## Common failure codes when deploying a disk pool

CODE	DESCRIPTION
UnexpectedError	Usually occurs when a backend unrecoverable error occurs. Retry the deployment. If the issue isn't resolved, contact Azure Support and provide the tracking ID of the error message.
DeploymentFailureZonalAllocationFailed	This occurs when Azure runs out of capacity to provision a VM in the specified region/zone. Retry the deployment at another time.
DeploymentFailureQuotaExceeded	The subscription used to deploy the disk pool is out of VM core quota in this region. You can <a href="#">request an increase in vCPU quota limits per Azure VM series</a> for Dsv3 series.
DeploymentFailurePolicyViolation	A policy on the subscription prevented the deployment of Azure resources that are required to support a disk pool. See the error for more details.
DeploymentTimeout	This occurs when the deployment of the disk pool infrastructure gets stuck and doesn't complete in the allotted time. Retry the deployment. If the issue persists, contact Azure support and provide the tracking ID of the error message.

CODE	DESCRIPTION
GoalStateApplicationTimeoutError	Occurs when the disk pool infrastructure stops responding to the resource provider. Confirm you meet the <a href="#">networking prerequisites</a> and then retry the deployment. If the issue persists, contact Azure support and provide the tracking ID of the error.
OngoingOperationInProgress	An ongoing operation is in-progress on the disk pool. Wait until that operation completes, then retry deployment.

## Common failure codes when enabling iSCSI on disk pools

CODE	DESCRIPTION
GoalStateApplicationError	Occurs when the iSCSI target configuration is invalid and cannot be applied to the disk pool. Retry the deployment. If the issue persists, contact Azure support and provide the tracking ID of the error.
GoalStateApplicationTimeoutError	Occurs when the disk pool infrastructure stops responding to the resource provider. Confirm you meet the <a href="#">networking prerequisites</a> and then retry the deployment. If the issue persists, contact Azure support and provide the tracking ID of the error.
OngoingOperationInProgress	An ongoing operation is in-progress on the disk pool. Wait until that operation completes, then retry deployment.

## Next steps

[Manage a disk pool \(preview\)](#)

# Overview of managed disk encryption options

9/21/2022 • 2 minutes to read • [Edit Online](#)

There are several types of encryption available for your managed disks, including Azure Disk Encryption (ADE), Server-Side Encryption (SSE) and encryption at host.

- **Azure Disk Encryption** helps protect and safeguard your data to meet your organizational security and compliance commitments. ADE encrypts the OS and data disks of Azure virtual machines (VMs) inside your VMs by using the [DM-Crypt](#) feature of Linux or the [BitLocker](#) feature of Windows. ADE is integrated with Azure Key Vault to help you control and manage the disk encryption keys and secrets. For full details, see [Azure Disk Encryption for Linux VMs](#) or [Azure Disk Encryption for Windows VMs](#).
- **Azure Disk Storage Server-Side Encryption** (also referred to as encryption-at-rest or Azure Storage encryption) automatically encrypts data stored on Azure managed disks (OS and data disks) when persisting on the Storage Clusters. When configured with a Disk Encryption Set (DES), it supports customer-managed keys as well. For full details, see [Server-side encryption of Azure Disk Storage](#).
- **Encryption at host** ensures that data stored on the VM host hosting your VM is encrypted at rest and flows encrypted to the Storage clusters. For full details, see [Encryption at host - End-to-end encryption for your VM data](#).
- **Confidential disk encryption** binds disk encryption keys to the virtual machine's TPM and makes the protected disk content accessible only to the VM. The TPM and VM guest state is always encrypted in attested code using keys released by a secure protocol that bypasses the hypervisor and host operating system. Currently only available for the OS disk. Encryption at host may be used for other disks on a Confidential VM in addition to Confidential Disk Encryption. For full details, see [DCasv5 and ECasv5 series confidential VMs](#).

Encryption is part of a layered approach to security and should be used with other recommendations to secure Virtual Machines and their disks. For full details, see [Security recommendations for virtual machines in Azure](#) and [Restrict import/export access to managed disks](#).

## Comparison

Here's a comparison of Disk Storage SSE, ADE, encryption at host, and Confidential disk encryption.

	ENCRYPTION AT REST (OS AND DATA DISKS)	TEMP DISK ENCRYPTION	ENCRYPTION OF CACHES	DATA FLOWS ENCRYPTED BETWEEN COMPUTE AND STORAGE	CUSTOMER CONTROL OF KEYS	DOES NOT USE YOUR VM'S CPU	WORKS FOR CUSTOM IMAGES	ENHANCED KEY PROTECTION	MICROSOFT DEFENDER FOR CLOUD DISK ENCRYPTION STATUS
Azure Disk Storage Server-Side Encryption at rest	✓	✗	✗	✗	✓ When configured with DES	✓	✓	✗	Unhealthy, not applicable if exempt
Azure Disk Encryption	✓	✓	✓	✓	✓	✗	✗ Does not work for custom Linux images	✗	Healthy
Encryption at Host	✓	✓	✓	✓	✓	✓	✓	✗	Unhealthy, not applicable if exempt
Confidential disk encryption	✓ For the OS disk only	✗	✓ For the OS disk only	✓ For the OS disk only	✓ For the OS disk only	✗	✓	✓	Unhealthy, not applicable if exempt

#### IMPORTANT

For Encryption at host and Confidential disk encryption, Microsoft Defender for Cloud does not detect the encryption state. We are in the process of updating Microsoft Defender

## Next steps

- [Azure Disk Encryption for Linux VMs](#)
- [Azure Disk Encryption for Windows VMs](#)
- [Server-side encryption of Azure Disk Storage](#)
- [Encryption at host](#)
- [DCav5 and ECav5 series confidential VMs](#)
- [Azure Security Fundamentals - Azure encryption overview](#)

# Server-side encryption of Azure Disk Storage

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Most Azure managed disks are encrypted with Azure Storage encryption, which uses server-side encryption (SSE) to protect your data and to help you meet your organizational security and compliance commitments. Azure Storage encryption automatically encrypts your data stored on Azure managed disks (OS and data disks) at rest by default when persisting it to the cloud. Disks with encryption at host enabled, however, are not encrypted through Azure Storage. For disks with encryption at host enabled, the server hosting your VM provides the encryption for your data, and that encrypted data flows into Azure Storage.

Data in Azure managed disks is encrypted transparently using 256-bit [AES encryption](#), one of the strongest block ciphers available, and is FIPS 140-2 compliant. For more information about the cryptographic modules underlying Azure managed disks, see [Cryptography API: Next Generation](#)

Azure Storage encryption does not impact the performance of managed disks and there is no additional cost. For more information about Azure Storage encryption, see [Azure Storage encryption](#).

## NOTE

Temporary disks are not managed disks and are not encrypted by SSE, unless you enable encryption at host.

## About encryption key management

You can rely on platform-managed keys for the encryption of your managed disk, or you can manage encryption using your own keys. If you choose to manage encryption with your own keys, you can specify a *customer-managed key* to use for encrypting and decrypting all data in managed disks.

The following sections describe each of the options for key management in greater detail.

### Platform-managed keys

By default, managed disks use platform-managed encryption keys. All managed disks, snapshots, images, and data written to existing managed disks are automatically encrypted-at-rest with platform-managed keys.

### Customer-managed keys

You can choose to manage encryption at the level of each managed disk, with your own keys. When you specify a customer-managed key, that key is used to protect and control access to the key that encrypts your data. Customer-managed keys offer greater flexibility to manage access controls.

You must use one of the following Azure key stores to store your customer-managed keys:

- [Azure Key Vault](#)
- [Azure Key Vault Managed Hardware Security Module \(HSM\)](#)

You can either import [your RSA keys](#) to your Key Vault or generate new RSA keys in Azure Key Vault. Azure managed disks handles the encryption and decryption in a fully transparent fashion using envelope encryption. It encrypts data using an [AES](#) 256 based data encryption key (DEK), which is, in turn, protected using your keys. The Storage service generates data encryption keys and encrypts them with customer-managed keys using RSA encryption. The envelope encryption allows you to rotate (change) your keys periodically as per your compliance policies without impacting your VMs. When you rotate your keys, the Storage service re-encrypts

the data encryption keys with the new customer-managed keys.

Managed Disks and the Key Vault or managed HSM must be in the same Azure region, but they can be in different subscriptions. They must also be in the same Azure Active Directory (Azure AD) tenant, unless you're using [Encrypt managed disks with cross-tenant customer-managed keys \(preview\)](#).

#### Full control of your keys

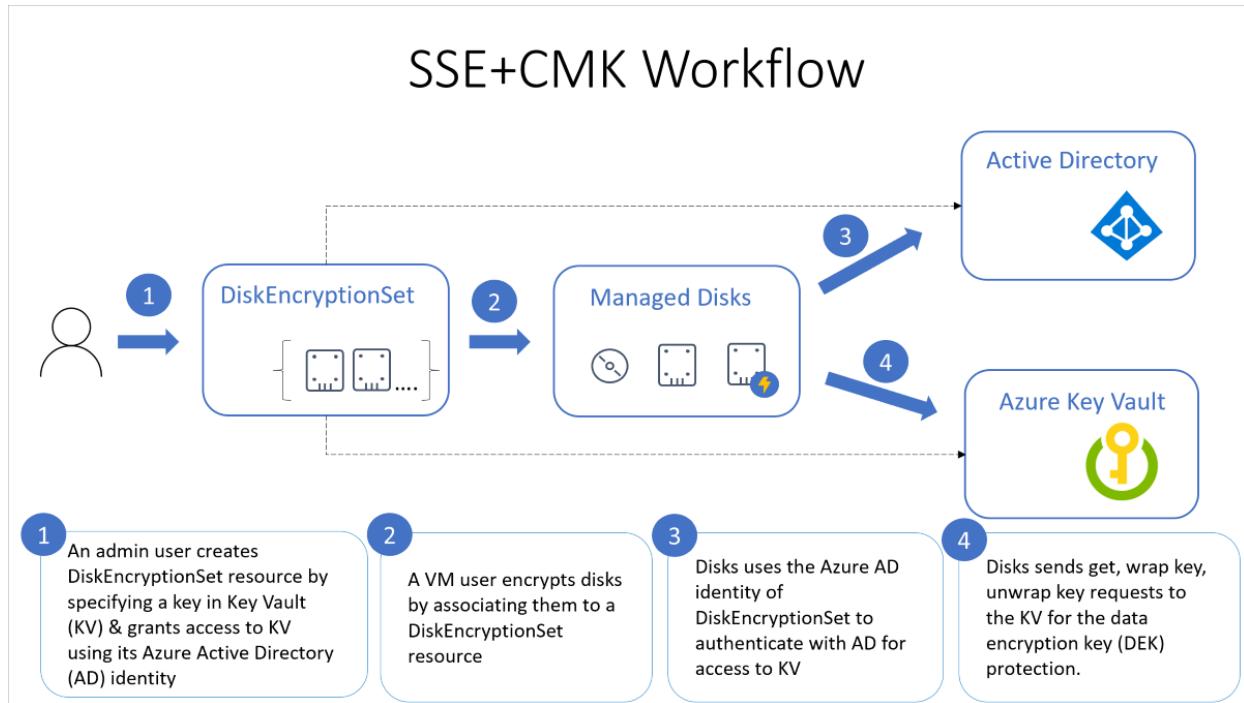
You must grant access to managed disks in your Key Vault or managed HSM to use your keys for encrypting and decrypting the DEK. This allows you full control of your data and keys. You can disable your keys or revoke access to managed disks at any time. You can also audit the encryption key usage with Azure Key Vault monitoring to ensure that only managed disks or other trusted Azure services are accessing your keys.

When a key is either disabled, deleted, or expired, any VMs with disks using that key will automatically shut down. After this, the VMs will not be usable unless the key is enabled again or you assign a new key.

#### NOTE

It is generally expected that Disk I/O (read or write operations) will start to fail 1 hour after a key is either disabled, deleted, or expired.

The following diagram shows how managed disks use Azure Active Directory and Azure Key Vault to make requests using the customer-managed key:



The following list explains the diagram in more detail:

1. An Azure Key Vault administrator creates key vault resources.
2. The key vault admin either imports their RSA keys to Key Vault or generate new RSA keys in Key Vault.
3. That administrator creates an instance of Disk Encryption Set resource, specifying an Azure Key Vault ID and a key URL. Disk Encryption Set is a new resource introduced for simplifying the key management for managed disks.
4. When a disk encryption set is created, a [system-assigned managed identity](#) is created in Azure Active Directory (AD) and associated with the disk encryption set.
5. The Azure key vault administrator then grants the managed identity permission to perform operations in the key vault.
6. A VM user creates disks by associating them with the disk encryption set. The VM user can also enable

server-side encryption with customer-managed keys for existing resources by associating them with the disk encryption set.

7. Managed disks use the managed identity to send requests to the Azure Key Vault.
8. For reading or writing data, managed disks sends requests to Azure Key Vault to encrypt (wrap) and decrypt (unwrap) the data encryption key in order to perform encryption and decryption of the data.

To revoke access to customer-managed keys, see [Azure Key Vault PowerShell](#) and [Azure Key Vault CLI](#). Revoking access effectively blocks access to all data in the storage account, as the encryption key is inaccessible by Azure Storage.

#### **Automatic key rotation of customer-managed keys**

You can choose to enable automatic key rotation to the latest key version. A disk references a key via its disk encryption set. When you enable automatic rotation for a disk encryption set, the system will automatically update all managed disks, snapshots, and images referencing the disk encryption set to use the new version of the key within one hour. To learn how to enable customer-managed keys with automatic key rotation, see [Set up an Azure Key Vault and DiskEncryptionSet with automatic key rotation](#).

#### **NOTE**

Virtual Machines will not be rebooted during automatic key rotation.

#### **Restrictions**

For now, customer-managed keys have the following restrictions:

- If this feature is enabled for your disk, you cannot disable it. If you need to work around this, you must copy all the data using either the [Azure PowerShell module](#) or the [Azure CLI](#), to an entirely different managed disk that isn't using customer-managed keys.
- Only [software](#) and [HSM RSA keys](#) of sizes 2,048-bit, 3,072-bit and 4,096-bit are supported, no other keys or sizes.
  - [HSM](#) keys require the [premium](#) tier of Azure Key vaults.
- Disks created from custom images that are encrypted using server-side encryption and customer-managed keys must be encrypted using the same customer-managed keys and must be in the same subscription.
- Snapshots created from disks that are encrypted with server-side encryption and customer-managed keys must be encrypted with the same customer-managed keys.
- Most resources related to your customer-managed keys (disk encryption sets, VMs, disks, and snapshots) must be in the same subscription and region.
  - Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.
- Disks, snapshots, and images encrypted with customer-managed keys cannot move to another resource group and subscription.
- Managed disks currently or previously encrypted using Azure Disk Encryption cannot be encrypted using customer-managed keys.
- Can only create up to 1000 disk encryption sets per region per subscription.
- For information about using customer-managed keys with shared image galleries, see [Preview: Use customer-managed keys for encrypting images](#).

#### **Supported regions**

Customer-managed keys are available in all regions that managed disks are available.

## **IMPORTANT**

Customer-managed keys rely on managed identities for Azure resources, a feature of Azure Active Directory (Azure AD). When you configure customer-managed keys, a managed identity is automatically assigned to your resources under the covers. If you subsequently move the subscription, resource group, or managed disk from one Azure AD directory to another, the managed identity associated with managed disks isn't transferred to the new tenant, so customer-managed keys may no longer work. For more information, see [Transferring a subscription between Azure AD directories](#).

To enable customer-managed keys for managed disks, see our articles covering how to enable it with either the [Azure PowerShell module](#), the [Azure CLI](#) or the [Azure portal](#).

## Encryption at host - End-to-end encryption for your VM data

When you enable encryption at host, that encryption starts on the VM host itself, the Azure server that your VM is allocated to. The data for your temporary disk and OS/data disk caches are stored on that VM host. After enabling encryption at host, all this data is encrypted at rest and flows encrypted to the Storage service, where it is persisted. Essentially, encryption at host encrypts your data from end-to-end. Encryption at host does not use your VM's CPU and doesn't impact your VM's performance.

Temporary disks and ephemeral OS disks are encrypted at rest with platform-managed keys when you enable end-to-end encryption. The OS and data disk caches are encrypted at rest with either customer-managed or platform-managed keys, depending on the selected disk encryption type. For example, if a disk is encrypted with customer-managed keys, then the cache for the disk is encrypted with customer-managed keys, and if a disk is encrypted with platform-managed keys then the cache for the disk is encrypted with platform-managed keys.

### **Restrictions**

- Doesn't support ultra disks.
- Cannot be enabled if Azure Disk Encryption (guest-VM encryption using bitlocker/DM-Crypt) is enabled on your VMs/virtual machine scale sets.
- Azure Disk Encryption cannot be enabled on disks that have encryption at host enabled.
- The encryption can be enabled on existing virtual machine scale set. However, only new VMs created after enabling the encryption are automatically encrypted.
- Existing VMs must be deallocated and reallocated in order to be encrypted.
- Supports ephemeral OS disks but only with platform-managed keys.

### **Supported VM sizes**

The complete list of supported VM sizes can be pulled programmatically. To learn how to retrieve them programmatically, refer to the finding supported VM sizes section of either the [Azure PowerShell module](#) or [Azure CLI](#) articles.

To enable end-to-end encryption using encryption at host, see our articles covering how to enable it with either the [Azure PowerShell module](#), the [Azure CLI](#), or the [Azure portal](#).

## Double encryption at rest

High security sensitive customers who are concerned of the risk associated with any particular encryption algorithm, implementation, or key being compromised can now opt for additional layer of encryption using a different encryption algorithm/mode at the infrastructure layer using platform managed encryption keys. This new layer can be applied to persisted OS and data disks, snapshots, and images, all of which will be encrypted at rest with double encryption.

### **Supported regions**

Double encryption is available in all regions that managed disks are available.

To enable double encryption at rest for managed disks, see our articles covering how to enable it with either the [Azure PowerShell module](#), the [Azure CLI](#) or the [Azure portal](#).

## Server-side encryption versus Azure disk encryption

[Azure Disk Encryption](#) leverages either the [DM-Crypt](#) feature of Linux or the [BitLocker](#) feature of Windows to encrypt managed disks with customer-managed keys within the guest VM. Server-side encryption with customer-managed keys improves on ADE by enabling you to use any OS types and images for your VMs by encrypting data in the Storage service.

### IMPORTANT

Customer-managed keys rely on managed identities for Azure resources, a feature of Azure Active Directory (Azure AD). When you configure customer-managed keys, a managed identity is automatically assigned to your resources under the covers. If you subsequently move the subscription, resource group, or managed disk from one Azure AD directory to another, the managed identity associated with managed disks is not transferred to the new tenant, so customer-managed keys may no longer work. For more information, see [Transferring a subscription between Azure AD directories](#).

## Next steps

- Enable end-to-end encryption using encryption at host with either the [Azure PowerShell module](#), the [Azure CLI](#), or the [Azure portal](#).
- Enable double encryption at rest for managed disks with either the [Azure PowerShell module](#), the [Azure CLI](#) or the [Azure portal](#).
- Enable customer-managed keys for managed disks with either the [Azure PowerShell module](#), the [Azure CLI](#) or the [Azure portal](#).
- [Explore the Azure Resource Manager templates for creating encrypted disks with customer-managed keys](#)
- [What is Azure Key Vault?](#)

# Use the Azure portal to enable server-side encryption with customer-managed keys for managed disks

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓

Azure Disk Storage allows you to manage your own keys when using server-side encryption (SSE) for managed disks, if you choose. For conceptual information on SSE with customer managed keys, as well as other managed disk encryption types, see the **Customer-managed keys** section of our disk encryption article: [Customer-managed keys](#)

## Restrictions

For now, customer-managed keys have the following restrictions:

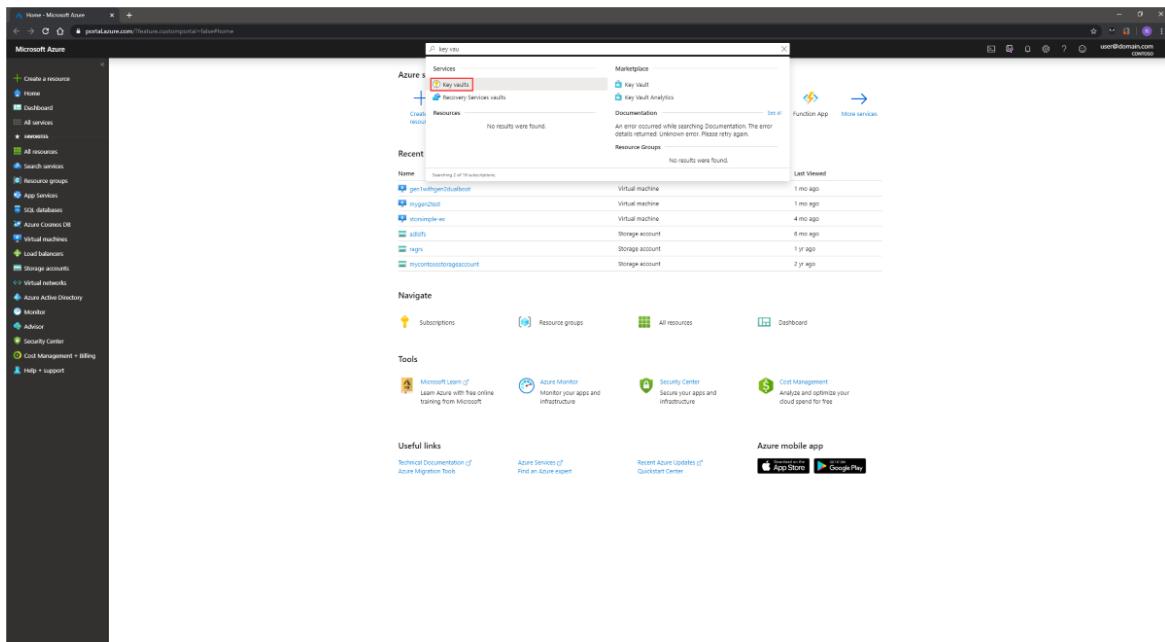
- If this feature is enabled for your disk, you cannot disable it. If you need to work around this, you must copy all the data to an entirely different managed disk that isn't using customer-managed keys:
  - For Linux: [Copy a managed disk](#)
  - For Windows: [Copy a managed disk](#)
- Only **software and HSM RSA keys** of sizes 2,048-bit, 3,072-bit and 4,096-bit are supported, no other keys or sizes.
  - **HSM** keys require the **premium** tier of Azure Key vaults.
- Disks created from custom images that are encrypted using server-side encryption and customer-managed keys must be encrypted using the same customer-managed keys and must be in the same subscription.
- Snapshots created from disks that are encrypted with server-side encryption and customer-managed keys must be encrypted with the same customer-managed keys.
- Most resources related to your customer-managed keys (disk encryption sets, VMs, disks, and snapshots) must be in the same subscription and region.
  - Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.
- Disks, snapshots, and images encrypted with customer-managed keys cannot move to another resource group and subscription.
- Managed disks currently or previously encrypted using Azure Disk Encryption cannot be encrypted using customer-managed keys.
- Can only create up to 1000 disk encryption sets per region per subscription.
- For information about using customer-managed keys with shared image galleries, see [Preview: Use customer-managed keys for encrypting images](#).

The following sections cover how to enable and use customer-managed keys for managed disks:

Setting up customer-managed keys for your disks will require you to create resources in a particular order, if you're doing it for the first time. First, you will need to create and set up an Azure Key Vault.

## Set up your Azure Key Vault

1. Sign into the Azure portal.
2. Search for and select Key Vaults.



### IMPORTANT

Your disk encryption set, VM, disks, and snapshots must all be in the same region and subscription for deployment to succeed. Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.

3. Select **+Create** to create a new Key Vault.
4. Create a new resource group.
5. Enter a key vault name, select a region, and select a pricing tier.

### NOTE

When creating the Key Vault instance, you must enable soft delete and purge protection. Soft delete ensures that the Key Vault holds a deleted key for a given retention period (90 day default). Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

6. Select **Review + Create**, verify your choices, then select **Create**.

Dashboard > Key vaults > Create key vault

## Create key vault

**Basics** [Access policy](#) [Networking](#) [Tags](#) [Review + create](#)

Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance. [Learn more](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

**Subscription \*** My Example Subscription

**Resource group \*** Select existing... [Create new](#)

**Instance details**

**Key vault name \*** [\(i\)](#)

**Region \***

**Pricing tier \*** [\(i\)](#)

**Review + create** [< Previous](#) [OK](#) [Cancel](#)

7. Once your key vault finishes deploying, select it.

8. Select **Keys** under **Settings**.

9. Select **Generate/Import**.

**my-example-vault - Keys**

[Search \(Ctrl+ /\)](#) [Generate/Import](#) [Refresh](#) [Restore Backup](#)

**Overview** **Activity log** **Access control (IAM)** **Tags** **Diagnose and solve problems** **Events (preview)**

**Settings**

- Keys** (highlighted with a red box)
- Secrets**
- Certificates**

**Name**  
There are no keys available.

10. Leave both **Key Type** set to RSA and **RSA Key Size** set to 2048.

11. Fill in the remaining selections as you like and then select **Create**.

The screenshot shows the 'Create a key' configuration page. It includes fields for Name (with a required asterisk), Key Type (RSA selected), RSA Key Size (2048 selected), and options for activation and expiration dates. The 'Enabled?' switch is set to 'Yes'. A large red 'Create' button is at the bottom.

**Create a key**

Options  
Generate

Name \* ⓘ

Key Type ⓘ  
RSA EC

RSA Key Size  
2048 3072 4096

Set activation date? ⓘ

Set expiration date? ⓘ

Enabled?  
Yes No

**Create**

### Add an Azure RBAC role

Now that you've created the Azure key vault and a key, you must add an Azure RBAC role, so you can use your Azure key vault with your disk encryption set.

1. Select **Access control (IAM)** and add a role.
2. Add either the **Key Vault Administrator**, **Owner**, or **Contributor** roles.

### Set up your disk encryption set

1. Search for **Disk Encryption Sets** and select it.
2. On the **Disk Encryption Sets** pane select **+Create**.
3. Select your resource group, name your encryption set, and select the same region as your key vault.
4. For **SSE Encryption type**, select **Encryption at-rest with a customer-managed key**.

**NOTE**

Once you create a disk encryption set with a particular encryption type, it cannot be changed. If you want to use a different encryption type, you must create a new disk encryption set.

5. Select **Click to select a key**.
6. Select the key vault and key you created previously, and the version.
7. Press **Select**.
8. If you want to enable [automatic rotation of customer managed keys](#), select **Auto key rotation**.
9. Select **Review + Create** and then **Create**.

## Create a disk encryption set

Basics Tags Review + create

Disk encryption sets allow you to manage encryption keys using server-side encryption for Standard HDD, Standard SSD, and Premium SSD managed disks. It will give you control of the encryption keys to meet your security and compliance needs in a few clicks. [Learn more about disk encryption sets.](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

My Example Subscription

Resource group \* ⓘ

[Create new](#)

### Instance details

Disk encryption set name \*

Region \* ⓘ

(US) South Central US

SSE encryption type \* ⓘ

Encryption at-rest with a customer-managed key

Key Vault \* ⓘ

Select a key vault

[Create new](#)

Key \* ⓘ

Select a key

Version \* ⓘ

Select a key version

Auto key rotation ⓘ

10. Navigate to the disk encryption set once it is deployed, and select the displayed alert.

The screenshot shows a deployment blade with a search bar and a delete button. Below the search bar are three tabs: Overview (selected), Activity log, and Overview. A red box highlights a message: "To associate a disk, image, or snapshot with this disk encryption set, you must grant permissions to the key vault my-example-key-vault. →".

11. This will grant your key vault permissions to the disk encryption set.

The screenshot shows a modal window with a green checkmark icon and the text "Successfully granted permissions". Below it says "Successfully granted permissions to the key vault 'my-example-key-vault'. a few seconds ago".

## Deploy a VM

Now that you've created and set up your key vault and the disk encryption set, you can deploy a VM using the encryption. The VM deployment process is similar to the standard deployment process, the only differences are that you need to deploy the VM in the same region as your other resources and you opt to use a customer managed key.

1. Search for **Virtual Machines** and select **+ Add** to create a VM.
2. On the **Basic** blade, select the same region as your disk encryption set and Azure Key Vault.
3. Fill in the other values on the **Basic** blade as you like.

## Create a virtual machine

[Basics](#) [Disks](#) [Networking](#) [Management](#) [Advanced](#) [Tags](#) [Review + create](#)

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image.

Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization.

Looking for classic VMs? [Create VM from Azure Marketplace](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

My Example Subscription

Resource group \* ⓘ

exampleresourcegroup

[Create new](#)

### Instance details

Virtual machine name \* ⓘ

examplevmname ✓

Region \* ⓘ

(US) South Central US

Availability options ⓘ

No infrastructure redundancy required

Image \* ⓘ

Windows Server 2019 Datacenter

[Browse all public and private images](#)

Azure Spot instance ⓘ

Yes  No

Size \* ⓘ

**Standard DS1 v2**

1 vcpu, 3.5 GiB memory

[Change size](#)

[Review + create](#)

< Previous

Next : Disks >

4. On the **Disks** blade, select **Encryption at rest with a customer-managed key**.
5. Select your disk encryption set in the **Disk encryption set** drop-down.
6. Make the remaining selections as you like.

Dashboard > Virtual machines > Create a virtual machine

## Create a virtual machine

**Basics** **Disks** **Networking** **Management** **Advanced** **Tags** **Review + create**

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

**Disk options**

OS disk type \* ⓘ  Premium SSD

Encryption options  Encryption at rest with a platform-managed key  Encryption at rest with a customer-managed key

**⚠️** Once a customer-managed key is used, you can't change the selection back to a platform-managed key. [Learn more about disk encryption.](#)

**Disk encryption set \***  exampleSetName

Enable Ultra Disk compatibility ⓘ  Yes  No  
Ultra Disk compatibility is not available for this VM size and location.

**Data disks**

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

LUN	Name	Size (GiB)	Disk type	Host caching
Create and attach a new disk <a href="#">Attach an existing disk</a>				

**Advanced**

[Review + create](#) [< Previous](#) [Next : Networking >](#)

## Enable on an existing disk

### Caution

Enabling disk encryption on any disks attached to a VM will require that you stop the VM.

1. Navigate to a VM that is in the same region as one of your disk encryption sets.
2. Open the VM and select Stop.

my-example-vm  
Virtual machine

Search (Ctrl+/  
Connect Start Restart Stop Capture Delete Refresh

Overview  
Activity log  
Access control (IAM)

Resource group (change) : southcentralus  
Status : Running  
Location : South Central US  
Subscription (change) : My Example Subscription

3. After the VM has finished stopping, select Disks and then select the disk you want to encrypt.

my-example-vm - Disks

Virtual machine

Search (Ctrl+ /)

Edit Refresh Encryption Swap OS Disk

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

Networking Disks (selected)

Size Security Extensions Continuous delivery (Preview) Availability + scaling

Managed disks created since June 10, 2017 are encrypted at rest with Storage Service Encryption

Ultra Disk compatibility is not available for this location.

Disk settings

Enable Ultra Disk compatibility ⓘ

Yes  No

OS disk

Name	Size
my-example-vm_OsDisk_1_c6be1c817df34ea8bc60e4ef70...	127 GiB

Data disks

None

+ Add data disk

4. Select **Encryption** and select **Encryption at rest with a customer-managed key** and then select your disk encryption set in the drop-down list.

5. Select **Save**.

my-example-vm\_OsDisk\_1\_c6be1c817df34ea8bc60e4ef70404870 - Encryption

Save Discard

Encryption

Encryption at rest with a platform-managed key  
 Encryption at rest with a customer-managed key

⚠ Once a customer-managed key is used, you can't change the selection back to a platform-managed key.  
Learn more about disk encryption.

Disk encryption set \*

exampleSetName

6. Repeat this process for any other disks attached to the VM you'd like to encrypt.

7. When your disks finish switching over to customer-managed keys, if there are no other attached disks you'd like to encrypt, you may start your VM.

#### IMPORTANT

Customer-managed keys rely on managed identities for Azure resources, a feature of Azure Active Directory (Azure AD). When you configure customer-managed keys, a managed identity is automatically assigned to your resources under the covers. If you subsequently move the subscription, resource group, or managed disk from one Azure AD directory to another, the managed identity associated with the managed disks is not transferred to the new tenant, so customer-managed keys may no longer work. For more information, see [Transferring a subscription between Azure AD directories](#).

#### Enable automatic key rotation on an existing disk encryption set

1. Navigate to the disk encryption set that you want to enable **automatic key rotation** on.
2. Under **Settings**, select **Key**.

3. Select **Auto key rotation** and select **Save**.

## Next steps

- Explore the Azure Resource Manager templates for creating encrypted disks with customer-managed keys
- What is Azure Key Vault?
- Replicate machines with customer-managed keys enabled disks
- Set up disaster recovery of VMware VMs to Azure with PowerShell
- Set up disaster recovery to Azure for Hyper-V VMs using PowerShell and Azure Resource Manager

# Azure PowerShell - Enable customer-managed keys with server-side encryption - managed disks

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure Disk Storage allows you to manage your own keys when using server-side encryption (SSE) for managed disks, if you choose. For conceptual information on SSE with customer-managed keys, and other managed disk encryption types, see the [Customer-managed keys](#) section of our disk encryption article.

## Restrictions

For now, customer-managed keys have the following restrictions:

- If this feature is enabled for your disk, you cannot disable it. If you need to work around this, you must [copy all the data](#) to an entirely different managed disk that isn't using customer-managed keys.
- Only [software and HSM RSA keys](#) of sizes 2,048-bit, 3,072-bit and 4,096-bit are supported, no other keys or sizes.
  - [HSM](#) keys require the [premium](#) tier of Azure Key vaults.
- Disks created from custom images that are encrypted using server-side encryption and customer-managed keys must be encrypted using the same customer-managed keys and must be in the same subscription.
- Snapshots created from disks that are encrypted with server-side encryption and customer-managed keys must be encrypted with the same customer-managed keys.
- Most resources related to your customer-managed keys (disk encryption sets, VMs, disks, and snapshots) must be in the same subscription and region.
  - Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.
- Disks, snapshots, and images encrypted with customer-managed keys cannot move to another resource group and subscription.
- Managed disks currently or previously encrypted using Azure Disk Encryption cannot be encrypted using customer-managed keys.
- Can only create up to 1000 disk encryption sets per region per subscription.
- For information about using customer-managed keys with shared image galleries, see [Preview: Use customer-managed keys for encrypting images](#).

## Set up an Azure Key Vault and DiskEncryptionSet optionally with automatic key rotation

To use customer-managed keys with SSE, you must set up an Azure Key Vault and a DiskEncryptionSet resource.

1. Make sure that you have installed latest [Azure PowerShell version](#), and you are signed in to an Azure account in with Connect-AzAccount
2. Create an instance of Azure Key Vault and encryption key.

When creating the Key Vault instance, you must enable purge protection. Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for

encrypting managed disks.

```
$ResourceGroupName="yourResourceGroupName"  
$LocationName="westcentralus"  
$keyVaultName="yourKeyVaultName"  
$keyName="yourKeyName"  
$keyDestination="Software"  
$diskEncryptionSetName="yourDiskEncryptionSetName"  
  
$keyVault = New-AzKeyVault -Name $keyVaultName `  
-ResourceGroupName $ResourceGroupName `  
-Location $LocationName `  
-EnablePurgeProtection  
  
$key = Add-AzKeyVaultKey -VaultName $keyVaultName `  
-Name $keyName `  
-Destination $keyDestination
```

3. Create an instance of a DiskEncryptionSet. You can set RotationToLatestKeyVersionEnabled equal to \$true to enable automatic rotation of the key. When you enable automatic rotation, the system will automatically update all managed disks, snapshots, and images referencing the disk encryption set to use the new version of the key within one hour.

```
$desConfig=New-AzDiskEncryptionSetConfig -Location $LocationName `  
-SourceVaultId $keyVault.ResourceId `  
-KeyUrl $key.Key.Kid `  
-IdentityType SystemAssigned `  
-RotationToLatestKeyVersionEnabled $false  
  
$des=New-AzDiskEncryptionSet -Name $diskEncryptionSetName `  
-ResourceGroupName $ResourceGroupName `  
-InputObject $desConfig
```

4. Grant the DiskEncryptionSet resource access to the key vault.

**NOTE**

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ObjectId $des.Identity.PrincipalId -  
PermissionsToKeys wrapkey,unwrapkey,get
```

### Use a key vault in a different subscription

Alternatively, you can manage your Azure Key Vaults centrally from a single subscription, and use the keys stored in the Key Vault to encrypt managed disks and snapshots in other subscriptions in your organization. This allows your security team to enforce and easily manage a robust security policy to a single subscription.

**IMPORTANT**

For this configuration, both your Key Vault and your disk encryption set must be in the same region and be using the same tenant.

The following script is an example of how you would configure a disk encryption set to use a key from a Key Vault in a different subscription, but same region:

```

$sourceSubscriptionId=<sourceSubID>
$sourceKeyVaultName=<sourceKVName>
$sourceKeyName=<sourceKeyName>

$targetSubscriptionId=<targetSubID>
$targetResourceGroupName=<targetRGName>
$targetDiskEncryptionSetName=<targetDiskEncSetName>
$location=<targetRegion>

Set-AzContext -Subscription $sourceSubscriptionId

$key = Get-AzKeyVaultKey -VaultName $sourceKeyVaultName -Name $sourceKeyName

Set-AzContext -Subscription $targetSubscriptionId

$desConfig=New-AzDiskEncryptionSetConfig -Location $location ` 
-KeyUrl $key.Key.Kid ` 
-IdentityType SystemAssigned ` 
-RotationToLatestKeyVersionEnabled $false

$des=New-AzDiskEncryptionSet -Name $targetDiskEncryptionSetName ` 
-ResourceGroupName $targetResourceGroupName ` 
-InputObject $desConfig

```

## Examples

Now that you've created and configured these resources, you can use them to secure your managed disks. The following are example scripts, each with a respective scenario, that you can use to secure your managed disks.

### **Create a VM using a Marketplace image, encrypting the OS and data disks with customer-managed keys**

Copy the script, replace all of the example values with your own parameters, and then run it.

```

$VMLocalAdminUser = "yourVMLocalAdminUserName"
$VMLocalAdminSecurePassword = ConvertTo-SecureString <password> -AsPlainText -Force
$LocationName = "yourRegion"
$ResourceGroupName = "yourResourceGroupName"
$ComputerName = "yourComputerName"
$VMName = "yourVMName"
$VMSize = "yourVMSize"
$diskEncryptionSetName="yourdiskEncryptionSetName"

$NetworkName = "yourNetworkName"
$NICName = "yourNICName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix
$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet
$NIC = New-AzNetworkInterface -Name $NICName -ResourceGroupName $ResourceGroupName -Location $LocationName -
SubnetId $Vnet.Subnets[0].Id

$Credential = New-Object System.Management.Automation.PSCredential ($VMLocalAdminUser,
$VMLocalAdminSecurePassword);

$VirtualMachine = New-AzVMConfig -VMName $VMName -VMSize $VMSize
$VirtualMachine = Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -
Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
$VirtualMachine = Add-AzVMNetworkInterface -VM $VirtualMachine -Id $NIC.Id
$VirtualMachine = Set-AzVMSourceImage -VM $VirtualMachine -PublisherName 'MicrosoftWindowsServer' -Offer
'WindowsServer' -Skus '2012-R2-Datacenter' -Version latest

$diskEncryptionSet=Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name
$diskEncryptionSetName

$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name $($VMName +"_OSDisk") -DiskEncryptionSetId
$diskEncryptionSet.Id -CreateOption FromImage

$VirtualMachine = Add-AzVMDataDisk -VM $VirtualMachine -Name $($VMName +"DataDisk1") -DiskSizeInGB 128 -
StorageAccountType Premium_LRS -CreateOption Empty -Lun 0 -DiskEncryptionSetId $diskEncryptionSet.Id

New-AzVM -ResourceGroupName $ResourceGroupName -Location $LocationName -VM $VirtualMachine -Verbose

```

## Create an empty disk encrypted using server-side encryption with customer-managed keys and attach it to a VM

Copy the script, replace all of the example values with your own parameters, and then run it.

```

$vmName = "yourVMName"
$LocationName = "westcentralus"
$ResourceGroupName = "yourResourceGroupName"
$diskName = "yourDiskName"
$diskSKU = "Premium_LRS"
$diskSizeinGiB = 30
$diskLUN = 1
$diskEncryptionSetName="yourDiskEncryptionSetName"

$vm = Get-AzVM -Name $vmName -ResourceGroupName $ResourceGroupName

$diskEncryptionSet=Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name
$diskEncryptionSetName

$vm = Add-AzVMDataDisk -VM $vm -Name $diskName -CreateOption Empty -DiskSizeInGB $diskSizeinGiB -
StorageAccountType $diskSKU -Lun $diskLUN -DiskEncryptionSetId $diskEncryptionSet.Id

Update-AzVM -ResourceGroupName $ResourceGroupName -VM $vm

```

## Encrypt existing managed disks

Your existing disks must not be attached to a running VM in order for you to encrypt them using the following script:

```

$rgName = "yourResourceGroupName"
$diskName = "yourDiskName"
$diskEncryptionSetName = "yourDiskEncryptionSetName"

$diskEncryptionSet = Get-AzDiskEncryptionSet -ResourceGroupName $rgName -Name $diskEncryptionSetName

New-AzDiskUpdateConfig -EncryptionType "EncryptionAtRestWithCustomerKey" -DiskEncryptionSetId
$diskEncryptionSet.Id | Update-AzDisk -ResourceGroupName $rgName -DiskName $diskName

```

## Encrypt an existing virtual machine scale set (uniform orchestration mode) with SSE and customer-managed keys

This script will work for scale sets in uniform orchestration mode only. For scale sets in flexible orchestration mode, follow the Encrypt existing managed disks for each VM.

Copy the script, replace all the example values with your own parameters, and then run it:

```

#set variables
$vmssname = "name of the vmss that is already created"
$diskencryptionsetname = "name of the diskencryptionset already created"
$vmssrgname = "vmss resourcegroup name"
$diskencryptionsetrgname = "diskencryptionset resourcegroup name"

#get vmss object and create diskencryptionset object attach to vmss os disk
$ssevmss = get-azvmss -ResourceGroupName $vmssrgname -VMScaleSetName $vmssname
$ssevmss.VirtualMachineProfile.StorageProfile.OsDisk.ManagedDisk.DiskEncryptionSet = New-Object -TypeName
Microsoft.Azure.Management.Compute.Models.DiskEncryptionSetParameters

#get diskencryption object and retrieve the resource id
$des = Get-AzDiskEncryptionSet -ResourceGroupName $diskencryptionsetrgname -Name $diskencryptionsetname
write-host "the diskencryptionset resource id is:" $des.Id

#associate DES resource id to os disk and update vmss
$ssevmss.VirtualMachineProfile.StorageProfile.OsDisk.ManagedDisk.DiskEncryptionSet.id = $des.Id
$ssevmss | update-azvmss

```

## Create a virtual machine scale set using a Marketplace image, encrypting the OS and data disks with

## customer-managed keys

Copy the script, replace all of the example values with your own parameters, and then run it.

```
$VMLocalAdminUser = "yourLocalAdminUser"
$VMLocalAdminSecurePassword = ConvertTo-SecureString Password@123 -AsPlainText -Force
$LocationName = "westcentralus"
$ResourceGroupName = "yourResourceGroupName"
$ComputerNamePrefix = "yourComputerNamePrefix"
$VMSScaleSetName = "yourVMSSName"
$VMSize = "Standard_DS3_v2"
$diskEncryptionSetName="yourDiskEncryptionSetName"

$NetworkName = "yourVNETName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix

$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet

$ipConfig = New-AzVmssIpConfig -Name "myIPConfig" -SubnetId $Vnet.Subnets[0].Id

$VMSS = New-AzVmssConfig -Location $LocationName -SkuCapacity 2 -SkuName $VMSSize -UpgradePolicyMode
'Automatic'

$VMSS = Add-AzVmssNetworkInterfaceConfiguration -Name "myVMSSNetworkConfig" -VirtualMachineScaleSet $VMSS -
Primary $true -IpConfiguration $ipConfig

$diskEncryptionSet=Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name
$diskEncryptionSetName

# Enable encryption at rest with customer managed keys for OS disk by setting DiskEncryptionSetId property

$VMSS = Set-AzVmssStorageProfile $VMSS -OsDiskCreateOption "FromImage" -DiskEncryptionSetId
$diskEncryptionSet.Id -ImageReferenceOffer 'WindowsServer' -ImageReferenceSku '2012-R2-Datacenter' -
ImageReferenceVersion latest -ImageReferencePublisher 'MicrosoftWindowsServer'

$VMSS = Set-AzVmssOsProfile $VMSS -ComputerNamePrefix $ComputerNamePrefix -AdminUsername $VMLocalAdminUser -
AdminPassword $VMLocalAdminSecurePassword

# Add a data disk encrypted at rest with customer managed keys by setting DiskEncryptionSetId property

$VMSS = Add-AzVmssDataDisk -VirtualMachineScaleSet $VMSS -CreateOption Empty -Lun 1 -DiskSizeGB 128 -
StorageAccountType Premium_LRS -DiskEncryptionSetId $diskEncryptionSet.Id

$Credential = New-Object System.Management.Automation.PSCredential ($VMLocalAdminUser,
$VMLocalAdminSecurePassword);

New-AzVmss -VirtualMachineScaleSet $VMSS -ResourceGroupName $ResourceGroupName -VMSScaleSetName
$VMSScaleSetName
```

## Change the key of a DiskEncryptionSet to rotate the key for all the resources referencing the DiskEncryptionSet

Copy the script, replace all of the example values with your own parameters, and then run it.

```
$ResourceGroupName="yourResourceGroupName"
$keyVaultName="yourKeyVaultName"
$keyName="yourKeyName"
$diskEncryptionSetName="yourDiskEncryptionSetName"

$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $ResourceGroupName

$keyVaultKey = Get-AzKeyVaultKey -VaultName $keyVaultName -Name $keyName

Update-AzDiskEncryptionSet -Name $diskEncryptionSetName -ResourceGroupName $ResourceGroupName -SourceVaultId
$keyVault.ResourceId -KeyUrl $keyVaultKey.Id
```

## Find the status of server-side encryption of a disk

```
$ResourceGroupName="yourResourceGroupName"
$DiskName="yourDiskName"

$disk=Get-AzDisk -ResourceGroupName $ResourceGroupName -DiskName $DiskName
$disk.Encryption.Type
```

### IMPORTANT

Customer-managed keys rely on managed identities for Azure resources, a feature of Azure Active Directory (Azure AD). When you configure customer-managed keys, a managed identity is automatically assigned to your resources under the covers. If you subsequently move the subscription, resource group, or managed disk from one Azure AD directory to another, the managed identity associated with the managed disks is not transferred to the new tenant, so customer-managed keys may no longer work. For more information, see [Transferring a subscription between Azure AD directories](#).

## Next steps

- [Explore the Azure Resource Manager templates for creating encrypted disks with customer-managed keys](#)
- [Replicate machines with customer-managed keys enabled disks](#)
- [Set up disaster recovery of VMware VMs to Azure with PowerShell](#)
- [Set up disaster recovery to Azure for Hyper-V VMs using PowerShell and Azure Resource Manager](#)

# Use the Azure CLI to enable server-side encryption with customer-managed keys for managed disks

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure Disk Storage allows you to manage your own keys when using server-side encryption (SSE) for managed disks, if you choose. For conceptual information on SSE with customer managed keys, as well as other managed disk encryption types, see the [Customer-managed keys](#) section of our disk encryption article.

## Restrictions

For now, customer-managed keys have the following restrictions:

- If this feature is enabled for your disk, you cannot disable it. If you need to work around this, you must [copy all the data](#) to an entirely different managed disk that isn't using customer-managed keys.
- Only [software and HSM RSA keys](#) of sizes 2,048-bit, 3,072-bit and 4,096-bit are supported, no other keys or sizes.
  - [HSM](#) keys require the [premium](#) tier of Azure Key vaults.
- Disks created from custom images that are encrypted using server-side encryption and customer-managed keys must be encrypted using the same customer-managed keys and must be in the same subscription.
- Snapshots created from disks that are encrypted with server-side encryption and customer-managed keys must be encrypted with the same customer-managed keys.
- Most resources related to your customer-managed keys (disk encryption sets, VMs, disks, and snapshots) must be in the same subscription and region.
  - Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.
- Disks, snapshots, and images encrypted with customer-managed keys cannot move to another resource group and subscription.
- Managed disks currently or previously encrypted using Azure Disk Encryption cannot be encrypted using customer-managed keys.
- Can only create up to 1000 disk encryption sets per region per subscription.
- For information about using customer-managed keys with shared image galleries, see [Preview: Use customer-managed keys for encrypting images](#).

## Create resources

Once the feature is enabled, you'll need to set up a DiskEncryptionSet and either an [Azure Key Vault](#) or an [Azure Key Vault Managed HSM](#).

### Azure Key Vault

- Install the latest [Azure CLI](#) and log to an Azure account in with [az login](#).
- Create an Azure Key Vault and encryption key.

When creating the Key Vault, you must enable purge protection. Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

## IMPORTANT

Don't camel case the region, if you do so, you may experience problems when assigning additional disks to the resource in the Azure portal.

```
subscriptionId=yourSubscriptionID
rgName=yourResourceGroupName
location=westcentralus
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName
diskName=yourDiskName

az account set --subscription $subscriptionId

az keyvault create -n $keyVaultName \
-g $rgName \
-l $location \
--enable-purge-protection true

az keyvault key create --vault-name $keyVaultName \
-n $keyName \
--protection software
```

- Create a DiskEncryptionSet. You can set enable-auto-key-rotation equal to true to enable automatic rotation of the key. When you enable automatic rotation, the system will automatically update all managed disks, snapshots, and images referencing the disk encryption set to use the new version of the key within one hour.

```
keyVaultKeyUrl=$(az keyvault key show --vault-name $keyVaultName --name $keyName --query [key.kid] -o tsv)

az disk-encryption-set create -n $diskEncryptionSetName
-l $location \
-g $rgName \
--key-url $keyVaultKeyUrl \
--enable-auto-key-rotation false
```

- Grant the DiskEncryptionSet resource access to the key vault.

## NOTE

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
desIdentity=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query
[identity.principalId] -o tsv)

az keyvault set-policy -n $keyVaultName \
-g $rgName \
--object-id $desIdentity \
--key-permissions wrapkey unwrapkey get
```

## Azure Key Vault Managed HSM

Alternatively, you can use a Managed HSM to handle your keys.

To do this, you must complete the following prerequisites:

- Install the latest [Azure CLI](#) and log in to an Azure account in with [az login](#).
- [Create and configure a managed HSM](#).
- [Assign permissions to a user, so they can manage your Managed HSM](#).

#### Configuration

Once you've created a Managed HSM and added permissions, enable purge protection and create an encryption key.

```
subscriptionId=yourSubscriptionID
rgName=yourResourceGroupName
location=westcentralus
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName
diskName=yourDiskName

az account set --subscription $subscriptionId

az keyvault update-hsm --subscription $subscriptionId -g $rgName --hsm-name $keyVaultName --enable-purge-protection true

az keyvault key create --hsm-name $keyVaultName --name $keyName --ops wrapKey unwrapKey --kty RSA-HSM --size 2048
```

Then, create a DiskEncryptionSet.

```
keyVaultKeyUrl=$(az keyvault key show --vault-name $keyVaultName --name $keyName --query [key.kid] -o tsv)

az disk-encryption-set create -n $diskEncryptionSetName
-l $location \
-g $rgName \
--key-url $keyVaultKeyUrl \
--enable-auto-key-rotation false
```

Finally, grant the DiskEncryptionSet access to the Managed HSM.

```
desIdentity=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [identity.principalId] -o tsv)

az keyvault role assignment create --hsm-name $keyVaultName --role "Managed HSM Crypto Service Encryption User" --assignee $desIdentity --scope /keys
```

Now that you've created and configured these resources, you can use them to secure your managed disks. The following links contain example scripts, each with a respective scenario, that you can use to secure your managed disks.

## Examples

[Create a VM using a Marketplace image, encrypting the OS and data disks with customer-managed keys](#)

```
rgName=yourResourceGroupName
vmName=yourVMName
location=westcentralus
vmSize=Standard_DS3_V2
image=UbuntuLTS
diskEncryptionSetName=yourDiskEncryptionSetName

diskEncryptionSetId=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [id] -o tsv)

az vm create -g $rgName -n $vmName -l $location --image $image --size $vmSize --generate-ssh-keys --os-disk-encryption-set $diskEncryptionSetId --data-disk-sizes-gb 128 128 --data-disk-encryption-sets $diskEncryptionSetId $diskEncryptionSetId
```

### Encrypt existing managed disks

Your existing disks must not be attached to a running VM in order for you to encrypt them using the following script:

```
rgName=yourResourceGroupName
diskName=yourDiskName
diskEncryptionSetName=yourDiskEncryptionSetName

az disk update -n $diskName -g $rgName --encryption-type EncryptionAtRestWithCustomerKey --disk-encryption-set $diskEncryptionSetId
```

### Create a virtual machine scale set using a Marketplace image, encrypting the OS and data disks with customer-managed keys

```
rgName=yourResourceGroupName
vmssName=yourVMSSName
location=westcentralus
vmSize=Standard_DS3_V2
image=UbuntuLTS
diskEncryptionSetName=yourDiskEncryptionSetName

diskEncryptionSetId=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [id] -o tsv)
az vmss create -g $rgName -n $vmssName --image UbuntuLTS --upgrade-policy automatic --admin-username azureuser --generate-ssh-keys --os-disk-encryption-set $diskEncryptionSetId --data-disk-sizes-gb 64 128 --data-disk-encryption-sets $diskEncryptionSetId $diskEncryptionSetId
```

### Create an empty disk encrypted using server-side encryption with customer-managed keys and attach it to a VM

```

vmName=yourVMName
rgName=yourResourceGroupName
diskName=yourDiskName
diskSkuName=Premium_LRS
diskSizeinGiB=30
location=westcentralus
diskLUN=2
diskEncryptionSetName=yourDiskEncryptionSetName

diskEncryptionSetId=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [id] -o tsv)

az disk create -n $diskName -g $rgName -l $location --encryption-type EncryptionAtRestWithCustomerKey --
disk-encryption-set $diskEncryptionSetId --size-gb $diskSizeinGiB --sku $diskSkuName

diskId=$(az disk show -n $diskName -g $rgName --query [id] -o tsv)

az vm disk attach --vm-name $vmName --lun $diskLUN --ids $diskId

```

## Change the key of a DiskEncryptionSet to rotate the key for all the resources referencing the DiskEncryptionSet

```

rgName=yourResourceGroupName
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName

keyVaultId=$(az keyvault show --name $keyVaultName --query [id] -o tsv)

keyVaultKeyUrl=$(az keyvault key show --vault-name $keyVaultName --name $keyName --query [key.kid] -o tsv)

az disk-encryption-set update -n keyrotationdes -g keyrotationtesting --key-url $keyVaultKeyUrl --source-
vault $keyVaultId

```

## Find the status of server-side encryption of a disk

```
az disk show -g yourResourceGroupName -n yourDiskName --query [encryption.type] -o tsv
```

### IMPORTANT

Customer-managed keys rely on managed identities for Azure resources, a feature of Azure Active Directory (Azure AD). When you configure customer-managed keys, a managed identity is automatically assigned to your resources under the covers. If you subsequently move the subscription, resource group, or managed disk from one Azure AD directory to another, the managed identity associated with the managed disks is not transferred to the new tenant, so customer-managed keys may no longer work. For more information, see [Transferring a subscription between Azure AD directories](#).

## Next steps

- Explore the Azure Resource Manager templates for creating encrypted disks with customer-managed keys
- Replicate machines with customer-managed keys enabled disks
- Set up disaster recovery of VMware VMs to Azure with PowerShell
- Set up disaster recovery to Azure for Hyper-V VMs using PowerShell and Azure Resource Manager

# Use the Azure portal to enable end-to-end encryption using encryption at host

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

When you enable encryption at host, data stored on the VM host is encrypted at rest and flows encrypted to the Storage service. For conceptual information on encryption at host, and other managed disk encryption types, see: [Encryption at host - End-to-end encryption for your VM data](#).

Temporary disks and ephemeral OS disks are encrypted at rest with platform-managed keys when you enable end-to-end encryption. The OS and data disk caches are encrypted at rest with either customer-managed or platform-managed keys, depending on what you select as the disk encryption type. For example, if a disk is encrypted with customer-managed keys, then the cache for the disk is encrypted with customer-managed keys, and if a disk is encrypted with platform-managed keys then the cache for the disk is encrypted with platform-managed keys.

## Restrictions

- Doesn't support ultra disks.
- Cannot be enabled if Azure Disk Encryption (guest-VM encryption using bitlocker/DM-Crypt) is enabled on your VMs/virtual machine scale sets.
- Azure Disk Encryption cannot be enabled on disks that have encryption at host enabled.
- The encryption can be enabled on existing virtual machine scale set. However, only new VMs created after enabling the encryption are automatically encrypted.
- Existing VMs must be deallocated and reallocated in order to be encrypted.
- Supports ephemeral OS disks but only with platform-managed keys.

## Supported VM sizes

Legacy VM Sizes are not supported. You can find the list of supported VM sizes by either using the [Azure PowerShell module](#) or [Azure CLI](#).

## Prerequisites

You must enable the feature for your subscription before you use the EncryptionAtHost property for your VM/VMSS. Follow the steps below to enable the feature for your subscription:

1. **Azure portal:** Select the Cloud Shell icon on the [Azure portal](#):



2. Execute the following command to register the feature for your subscription

- [Azure PowerShell](#)
- [Azure CLI](#)

```
Register-AzProviderFeature -FeatureName "EncryptionAtHost" -ProviderNamespace "Microsoft.Compute"
```

3. Confirm that the registration state is **Registered** (takes a few minutes) using the command below before trying out the feature.

- Azure PowerShell
- Azure CLI

```
Get-AzProviderFeature -FeatureName "EncryptionAtHost" -ProviderNamespace "Microsoft.Compute"
```

Sign in to the Azure portal using the [provided link](#).

#### IMPORTANT

You must use the [provided link](#) to access the Azure portal. Encryption at host is not currently visible in the public Azure portal without using the link.

## Deploy a VM with platform-managed keys

1. Sign in to the [Azure portal](#).
2. Search for **Virtual Machines** and select **+ Add** to create a VM.
3. Create a new virtual machine, select an appropriate region and a supported VM size.
4. Fill in the other values on the **Basic** pane as you like, then proceed to the  pane.

### Create a virtual machine

 Changing Basic options may reset selections you have made. Review all options prior to creating the virtual machine.

[Basics](#)   [Disks](#)   [Networking](#)   [Management](#)   [Advanced](#)   [Tags](#)   [Review + create](#)

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more](#)

#### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* 

My Example Subscription 

Resource group \* 

myexamplergroup 

[Create new](#)

#### Instance details

Virtual machine name \* 

my-example-vm-name 

Region \* 

(US) West US 2 

Availability options 

No infrastructure redundancy required 

Image \* 

Windows Server 2019 Datacenter 

[Browse all public and private images](#)

Azure Spot instance 

Yes  No

Size \* 

Standard\_DS2\_v2 - 2 vcpus, 7 GiB memory 

[Select size](#)

5. On the Disks pane, select **Encryption at host**.

6. Make the remaining selections as you like.

The screenshot shows the 'Disks' tab selected in the top navigation bar. Under 'Disk options', the 'OS disk type' is set to 'Premium SSD (locally-redundant storage)' and the 'SSE encryption type' is set to '(Default) Encryption at-rest with a platform-managed key'. A checkbox for 'Encryption at host' is checked. The overall message indicates that Azure VMs have one operating system disk and a temporary disk for short-term storage, with the size of the VM determining the type of storage allowed.

7. Finish the VM deployment process, make selections that fit your environment.

You have now deployed a VM with encryption at host enabled, and the cache for the disk is encrypted using platform-managed keys.

## Deploy a VM with customer-managed keys

Alternatively, you can use customer-managed keys to encrypt your disk caches.

### Create an Azure Key Vault and disk encryption set

Once the feature is enabled, you'll need to set up an Azure Key Vault and a disk encryption set, if you haven't already.

Setting up customer-managed keys for your disks will require you to create resources in a particular order, if you're doing it for the first time. First, you will need to create and set up an Azure Key Vault.

## Set up your Azure Key Vault

1. Sign into the [Azure portal](#).

2. Search for and select Key Vaults.

The screenshot shows the Azure portal search results for 'Key Vaults'. The search bar at the top has 'Key Vaults' typed into it. Below the search bar, there is a 'Services' section with a red box highlighting the 'Key Vault' item. Other services listed include Recovery Services vaults, Marketplace, Key Vault Analytics, Documentation, Function App, and More services. The 'Recent' section shows a list of recent resources, including 'gen-hvagen1Vault', 'myrgen1Vault', 'azurerm', 'adots', 'regis', and 'mycontosostorageaccount'. The 'Last Viewed' section shows 'Virtual machine' and 'Storage account' with their last viewed times. The bottom of the page features a 'Navigate' section with links to Microsoft Learn, Azure Monitor, Security Center, and Cost Management, along with 'Useful links' and 'Azure mobile app' sections.

### IMPORTANT

Your disk encryption set, VM, disks, and snapshots must all be in the same region and subscription for deployment to succeed. Azure Key Vaults may be used from a different subscription but must be in the same region and tenant as your disk encryption set.

3. Select **+Create** to create a new Key Vault.
4. Create a new resource group.
5. Enter a key vault name, select a region, and select a pricing tier.

### NOTE

When creating the Key Vault instance, you must enable soft delete and purge protection. Soft delete ensures that the Key Vault holds a deleted key for a given retention period (90 day default). Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

6. Select **Review + Create**, verify your choices, then select **Create**.

The screenshot shows the 'Create key vault' wizard interface. The 'Access policy' tab is selected. The 'Subscription' dropdown is set to 'My Example Subscription'. The 'Resource group' dropdown has 'Select existing...' and 'Create new' options. A tooltip for 'Resource group' explains it's a container for related resources. The 'Instance details' section includes fields for 'Key vault name', 'Region', and 'Pricing tier'. At the bottom, there are 'Review + create', '< Prev', 'OK', and 'Cancel' buttons.

7. Once your key vault finishes deploying, select it.
8. Select **Keys** under **Settings**.

9. Select **Generate/Import**.

The screenshot shows the 'my-example-vault - Keys' page in the Azure portal. On the left, there's a sidebar with icons for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Events (preview), Settings, Keys (which is highlighted with a red box), Secrets, and Certificates. At the top right, there are buttons for Generate/Import (highlighted with a red box), Refresh, and Restore Backup. The main area displays a table with one row: 'Name' and 'There are no keys available.'

10. Leave both **Key Type** set to RSA and **RSA Key Size** set to 2048.

11. Fill in the remaining selections as you like and then select **Create**.

The screenshot shows the 'Create a key' dialog. It has sections for Options (Generate selected), Name (empty input field), Key Type (RSA selected), RSA Key Size (2048 selected), Set activation date? (unchecked), Set expiration date? (unchecked), Enabled? (Yes selected), and a 'Create' button at the bottom (highlighted with a red box).

### Add an Azure RBAC role

Now that you've created the Azure key vault and a key, you must add an Azure RBAC role, so you can use your Azure key vault with your disk encryption set.

1. Select **Access control (IAM)** and add a role.
2. Add either the **Key Vault Administrator**, **Owner**, or **Contributor** roles.

# Set up your disk encryption set

1. Search for **Disk Encryption Sets** and select it.
2. On the **Disk Encryption Sets** pane select **+Create**.
3. Select your resource group, name your encryption set, and select the same region as your key vault.
4. For **SSE Encryption type**, select **Encryption at-rest with a customer-managed key**.

## NOTE

Once you create a disk encryption set with a particular encryption type, it cannot be changed. If you want to use a different encryption type, you must create a new disk encryption set.

5. Select **Click to select a key**.
6. Select the key vault and key you created previously, and the version.
7. Press **Select**.
8. If you want to enable **automatic rotation of customer managed keys**, select **Auto key rotation**.
9. Select **Review + Create** and then **Create**.

## Create a disk encryption set ...

[Basics](#)   [Tags](#)   [Review + create](#)

Disk encryption sets allow you to manage encryption keys using server-side encryption for Standard HDD, Standard SSD, and Premium SSD managed disks. It will give you control of the encryption keys to meet your security and compliance needs in a few clicks. [Learn more about disk encryption sets](#).

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

▼

Resource group \* ⓘ

▼

[Create new](#)

### Instance details

Disk encryption set name \*

Region \* ⓘ

▼

SSE encryption type \* ⓘ

▼

Key Vault \* ⓘ

▼

[Create new](#)

Key \* ⓘ

▼

Version \* ⓘ

▼

Auto key rotation ⓘ

10. Navigate to the disk encryption set once it is deployed, and select the displayed alert.

11. This will grant your key vault permissions to the disk encryption set.

Successfully granted permissions  
Successfully granted permissions to the key vault 'my-example-key-vault'.  
a few seconds ago

## Deploy a VM

Now that you've setup an Azure Key Vault and disk encryption set, you can deploy a VM and it will use encryption at host.

1. Sign in to the [Azure portal](#).
2. Search for **Virtual Machines** and select **+ Add** to create a VM.
3. Create a new virtual machine, select an appropriate region and a supported VM size.
4. Fill in the other values on the **Basic** pane as you like, then proceed to the **Disks** pane.

**Create a virtual machine**

**Basics**   **Disks**   **Networking**   **Management**   **Advanced**   **Tags**   **Review + create**

⚠️ Changing Basic options may reset selections you have made. Review all options prior to creating the virtual machine.

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more ↗](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

**Subscription** \* ⓘ My Example Subscription

**Resource group** \* ⓘ myexamplergroup [Create new](#)

**Instance details**

**Virtual machine name** \* ⓘ my-example-vm-name

**Region** \* ⓘ (US) West US 2

**Availability options** ⓘ No infrastructure redundancy required

**Image** \* ⓘ Windows Server 2019 Datacenter [Browse all public and private images](#)

**Azure Spot instance** ⓘ  Yes  No

**Size** \* ⓘ Standard\_DS2\_v2 - 2 vcpus, 7 GiB memory [Select size](#)

5. On the **Disks** pane, select **Encryption at-rest for customer-managed key** for SSE encryption type and select your disk encryption set.

6. Select **Encryption at host**.

7. Make the remaining selections as you like.

Disk options	
OS disk type *	Premium SSD (locally-redundant storage)
SSE encryption type *	Encryption at-rest with a customer-managed key
Disk encryption set *	example-set
Encryption at host	<input checked="" type="checkbox"/>

8. Finish the VM deployment process, make selections that fit your environment.

You have now deployed a VM with encryption at host enabled.

## Disable host based encryption

Make sure your VM is deallocated first, you cannot disable encryption at host unless your VM is deallocated.

1. On your VM, select **Disks** and then select **Additional settings**.

The screenshot shows the Azure portal's Disks blade for a virtual machine. The left sidebar has a 'Settings' section with icons for Networking, Connect, Disks (which is selected and highlighted in grey), Size, Security, and Advisor recommendations. The main area has tabs for 'Save', 'Discard', 'Refresh', and 'Additional settings' (which is highlighted with a red box). The 'OS disk' section shows a table with columns 'Storage type', 'Size (GiB)', and 'Max IOPS'. One row shows 'Premium SSD LRS', '30', and '120'. The 'Data disks' section shows a table with columns 'LUN', 'Disk name', and 'Storage type'. A note says 'No data disks attached'. At the bottom, there are buttons for 'Create and attach a new disk' and 'Attach existing disks'.

2. Select **No** for **Encryption at host** then select **Save**.

## Next steps

[Azure Resource Manager template samples](#)

# Use the Azure PowerShell module to enable end-to-end encryption using encryption at host

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

When you enable encryption at host, data stored on the VM host is encrypted at rest and flows encrypted to the Storage service. For conceptual information on encryption at host, as well as other managed disk encryption types, see [Encryption at host - End-to-end encryption for your VM data](#).

## Restrictions

- Doesn't support ultra disks.
- Cannot be enabled if Azure Disk Encryption (guest-VM encryption using bitlocker/DM-Crypt) is enabled on your VMs/virtual machine scale sets.
- Azure Disk Encryption cannot be enabled on disks that have encryption at host enabled.
- The encryption can be enabled on existing virtual machine scale set. However, only new VMs created after enabling the encryption are automatically encrypted.
- Existing VMs must be deallocated and reallocated in order to be encrypted.
- Supports ephemeral OS disks but only with platform-managed keys.

## Supported VM sizes

The complete list of supported VM sizes can be pulled programmatically. To learn how to retrieve them programmatically, refer to the [Finding supported VM sizes](#) section. Upgrading the VM size will result in validation to check if the new VM size supports the EncryptionAtHost feature.

## Prerequisites

You must enable the feature for your subscription before you use the EncryptionAtHost property for your VM/VMSS. Please follow the steps below to enable the feature for your subscription:

1. Execute the following command to register the feature for your subscription

```
Register-AzProviderFeature -FeatureName "EncryptionAtHost" -ProviderNamespace "Microsoft.Compute"
```

2. Please check that the registration state is Registered (takes a few minutes) using the command below before trying out the feature.

```
Get-AzProviderFeature -FeatureName "EncryptionAtHost" -ProviderNamespace "Microsoft.Compute"
```

## Create an Azure Key Vault and DiskEncryptionSet

Once the feature is enabled, you'll need to set up an Azure Key Vault and a DiskEncryptionSet, if you haven't already.

1. Make sure that you have installed latest [Azure PowerShell version](#), and you are signed in to an Azure account in with Connect-AzAccount
2. Create an instance of Azure Key Vault and encryption key.

When creating the Key Vault instance, you must enable purge protection. Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

```
$ResourceGroupName="yourResourceGroupName"
$LocationName="westcentralus"
$keyVaultName="yourKeyVaultName"
$keyName="yourKeyName"
$keyDestination="Software"
$diskEncryptionSetName="yourDiskEncryptionSetName"

$keyVault = New-AzKeyVault -Name $keyVaultName ` 
-ResourceGroupName $ResourceGroupName ` 
-Location $LocationName ` 
-EnablePurgeProtection

$key = Add-AzKeyVaultKey -VaultName $keyVaultName ` 
-Name $keyName ` 
-Destination $keyDestination
```

3. Create an instance of a DiskEncryptionSet. You can set RotationToLatestKeyVersionEnabled equal to \$true to enable automatic rotation of the key. When you enable automatic rotation, the system will automatically update all managed disks, snapshots, and images referencing the disk encryption set to use the new version of the key within one hour.

```
$desConfig=New-AzDiskEncryptionSetConfig -Location $LocationName ` 
-SourceVaultId $keyVault.ResourceId ` 
-KeyUrl $key.Key.Kid ` 
-IdentityType SystemAssigned ` 
-RotationToLatestKeyVersionEnabled $false

$des=New-AzDiskEncryptionSet -Name $diskEncryptionSetName ` 
-ResourceGroupName $ResourceGroupName ` 
-InputObject $desConfig
```

4. Grant the DiskEncryptionSet resource access to the key vault.

#### NOTE

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ObjectId $des.Identity.PrincipalId - 
PermissionsToKeys wrapkey,unwrapkey,get
```

### Use a key vault in a different subscription

Alternatively, you can manage your Azure Key Vaults centrally from a single subscription, and use the keys stored in the Key Vault to encrypt managed disks and snapshots in other subscriptions in your organization. This allows your security team to enforce and easily manage a robust security policy to a single subscription.

## IMPORTANT

For this configuration, both your Key Vault and your disk encryption set must be in the same region and be using the same tenant.

The following script is an example of how you would configure a disk encryption set to use a key from a Key Vault in a different subscription, but same region:

```
$sourceSubscriptionId=<sourceSubID>
$sourceKeyVaultName=<sourceKVName>
$sourceKeyName=<sourceKeyName>

$targetSubscriptionId=<targetSubID>
$targetResourceGroupName=<targetRGName>
$targetDiskEncryptionSetName=<targetDiskEncSetName>
$location=<targetRegion>

Set-AzContext -Subscription $sourceSubscriptionId

$key = Get-AzKeyVaultKey -VaultName $sourceKeyVaultName -Name $sourceKeyName

Set-AzContext -Subscription $targetSubscriptionId

$desConfig=New-AzDiskEncryptionSetConfig -Location $location ` 
-KeyUrl $key.Key.Kid ` 
-IdentityType SystemAssigned ` 
-RotationToLatestKeyVersionEnabled $false

$des=New-AzDiskEncryptionSet -Name $targetDiskEncryptionSetName ` 
-ResourceGroupName $targetResourceGroupName ` 
-InputObject $desConfig
```

## Enable encryption at host for disks attached to VM and virtual machine scale sets

You can enable encryption at host by setting a new property EncryptionAtHost under securityProfile of VMs or virtual machine scale sets using the API version **2020-06-01** and above.

```
"securityProfile": { "encryptionAtHost": "true" }
```

## Examples

### Create a VM with encryption at host enabled with customer-managed keys.

Create a VM with managed disks using the resource URI of the DiskEncryptionSet created earlier to encrypt cache of OS and data disks with customer-managed keys. The temp disks are encrypted with platform-managed keys.

```

$VMLocalAdminUser = "yourVMLocalAdminUserName"
$VMLocalAdminSecurePassword = ConvertTo-SecureString <password> -AsPlainText -Force
$LocationName = "yourRegion"
$ResourceGroupName = "yourResourceGroupName"
$ComputerName = "yourComputerName"
$VMName = "yourVMName"
$VMSize = "yourVMSize"
$diskEncryptionSetName="yourdiskEncryptionSetName"

$NetworkName = "yourNetworkName"
$NICName = "yourNICName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix
$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet
$NIC = New-AzNetworkInterface -Name $NICName -ResourceGroupName $ResourceGroupName -Location $LocationName -
SubnetId $Vnet.Subnets[0].Id

$Credential = New-Object System.Management.Automation.PSCredential ($VMLocalAdminUser,
$VMLocalAdminSecurePassword);

# Enable encryption at host by specifying EncryptionAtHost parameter

$VirtualMachine = New-AzVMConfig -VMName $VMName -VMSize $VMSize -EncryptionAtHost
$VirtualMachine = Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -
Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
$VirtualMachine = Add-AzVMNetworkInterface -VM $VirtualMachine -Id $NIC.Id
$VirtualMachine = Set-AzVMSourceImage -VM $VirtualMachine -PublisherName 'MicrosoftWindowsServer' -Offer
'WindowsServer' -Skus '2012-R2-Datacenter' -Version latest

$diskEncryptionSet=Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name
$diskEncryptionSetName

# Enable encryption with a customer managed key for OS disk by setting DiskEncryptionSetId property

$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name $($VMName + "_OSDisk") -DiskEncryptionSetId
$diskEncryptionSet.Id -CreateOption FromImage

# Add a data disk encrypted with a customer managed key by setting DiskEncryptionSetId property

$VirtualMachine = Add-AzVMDataDisk -VM $VirtualMachine -Name $($VMName + "DataDisk1") -DiskSizeInGB 128 -
StorageAccountType Premium_LRS -CreateOption Empty -Lun 0 -DiskEncryptionSetId $diskEncryptionSet.Id

New-AzVM -ResourceGroupName $ResourceGroupName -Location $LocationName -VM $VirtualMachine -Verbose

```

### Create a VM with encryption at host enabled with platform-managed keys.

Create a VM with encryption at host enabled to encrypt cache of OS/data disks and temp disks with platform-managed keys.

```

$VMLocalAdminUser = "yourVMLocalAdminUserName"
$VMLocalAdminSecurePassword = ConvertTo-SecureString <password> -AsPlainText -Force
$LocationName = "yourRegion"
$ResourceGroupName = "yourResourceGroupName"
$ComputerName = "yourComputerName"
$VMName = "yourVMName"
$VMSize = "yourVMSize"

$NetworkName = "yourNetworkName"
$NICName = "yourNICName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix
$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet
$NIC = New-AzNetworkInterface -Name $NICName -ResourceGroupName $ResourceGroupName -Location $LocationName -
SubnetId $Vnet.Subnets[0].Id

$Credential = New-Object System.Management.Automation.PSCredential ($VMLocalAdminUser,
$VMLocalAdminSecurePassword);

# Enable encryption at host by specifying EncryptionAtHost parameter

$VirtualMachine = New-AzVMConfig -VMName $VMName -VMSize $VMSize -EncryptionAtHost

$VirtualMachine = Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -
Credential $Credential -ProvisionVMAgent -EnableAutoUpdate
$VirtualMachine = Add-AzVMNetworkInterface -VM $VirtualMachine -Id $NIC.Id
$VirtualMachine = Set-AzVMSourceImage -VM $VirtualMachine -PublisherName 'MicrosoftWindowsServer' -Offer
'WindowsServer' -Skus '2012-R2-Datacenter' -Version latest

$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name $($VMName + "_OSDisk") -CreateOption FromImage

$VirtualMachine = Add-AzVMDataDisk -VM $VirtualMachine -Name $($VMName + "DataDisk1") -DiskSizeInGB 128 -
StorageAccountType Premium_LRS -CreateOption Empty -Lun 0

New-AzVM -ResourceGroupName $ResourceGroupName -Location $LocationName -VM $VirtualMachine

```

## Update a VM to enable encryption at host.

```

$ResourceGroupName = "yourResourceGroupName"
$VMName = "yourVMName"

$VM = Get-AzVM -ResourceGroupName $ResourceGroupName -Name $VMName

Stop-AzVM -ResourceGroupName $ResourceGroupName -Name $VMName -Force

Update-AzVM -VM $VM -ResourceGroupName $ResourceGroupName -EncryptionAtHost $true

```

## Check the status of encryption at host for a VM

```

$ResourceGroupName = "yourResourceGroupName"
$VMName = "yourVMName"

$VM = Get-AzVM -ResourceGroupName $ResourceGroupName -Name $VMName

$VM.SecurityProfile.EncryptionAtHost

```

## Disable encryption at host

You must deallocate your VM before you can disable encryption at host.

```

$ResourceGroupName = "yourResourceGroupName"
$VMName = "yourVMName"

$VM = Get-AzVM -ResourceGroupName $ResourceGroupName -Name $VMName

Stop-AzVM -ResourceGroupName $ResourceGroupName -Name $VMName -Force

Update-AzVM -VM $VM -ResourceGroupName $ResourceGroupName -EncryptionAtHost $false

```

### Create a virtual machine scale set with encryption at host enabled with customer-managed keys.

Create a virtual machine scale set with managed disks using the resource URI of the DiskEncryptionSet created earlier to encrypt cache of OS and data disks with customer-managed keys. The temp disks are encrypted with platform-managed keys.

```

$VMLocalAdminUser = "yourLocalAdminUser"
$VMLocalAdminSecurePassword = ConvertTo-SecureString Password@123 -AsPlainText -Force
$LocationName = "westcentralus"
$ResourceGroupName = "yourResourceGroupName"
$ComputerNamePrefix = "yourComputerNamePrefix"
$VMSScaleSetName = "yourVMSSName"
$VMSize = "Standard_DS3_v2"
$diskEncryptionSetName="yourDiskEncryptionSetName"

$NetworkName = "yourVNETName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix

$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet

$ipConfig = New-AzVmssIpConfig -Name "myIPConfig" -SubnetId $Vnet.Subnets[0].Id

# Enable encryption at host by specifying EncryptionAtHost parameter

$VMSS = New-AzVmssConfig -Location $LocationName -SkuCapacity 2 -SkuName $VMSize -UpgradePolicyMode
'Automatic' -EncryptionAtHost

$VMSS = Add-AzVmssNetworkInterfaceConfiguration -Name "myVMSSNetworkConfig" -VirtualMachineScaleSet $VMSS -
Primary $true -IpConfiguration $ipConfig

$diskEncryptionSet=Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name
$diskEncryptionSetName

# Enable encryption with a customer managed key for the OS disk by setting DiskEncryptionSetId property

$VMSS = Set-AzVmssStorageProfile $VMSS -OsDiskCreateOption "FromImage" -DiskEncryptionSetId
$diskEncryptionSet.Id -ImageReferenceOffer 'WindowsServer' -ImageReferenceSku '2012-R2-Datacenter' -
ImageReferenceVersion latest -ImageReferencePublisher 'MicrosoftWindowsServer'

$VMSS = Set-AzVmssOsProfile $VMSS -ComputerNamePrefix $ComputerNamePrefix -AdminUsername $VMLocalAdminUser -
AdminPassword $VMLocalAdminSecurePassword

# Add a data disk encrypted with a customer managed key by setting DiskEncryptionSetId property

$VMSS = Add-AzVmssDataDisk -VirtualMachineScaleSet $VMSS -CreateOption Empty -Lun 1 -DiskSizeGB 128 -
StorageAccountType Premium_LRS -DiskEncryptionSetId $diskEncryptionSet.Id

```

### Create a virtual machine scale set with encryption at host enabled with platform-managed keys.

Create a virtual machine scale set with encryption at host enabled to encrypt cache of OS/data disks and temp

disks with platform-managed keys.

```
$VMLocalAdminUser = "yourLocalAdminUser"
$VMLocalAdminSecurePassword = ConvertTo-SecureString Password@123 -AsPlainText -Force
$LocationName = "westcentralus"
$ResourceGroupName = "yourResourceGroupName"
$ComputerNamePrefix = "yourComputerNamePrefix"
$VMSScaleSetName = "yourVMSSName"
$VMSize = "Standard_DS3_v2"

$NetworkName = "yourVNETName"
$SubnetName = "yourSubnetName"
$SubnetAddressPrefix = "10.0.0.0/24"
$VnetAddressPrefix = "10.0.0.0/16"

$SingleSubnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix

$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location
$LocationName -AddressPrefix $VnetAddressPrefix -Subnet $SingleSubnet

$ipConfig = New-AzVmssIpConfig -Name "myIPConfig" -SubnetId $Vnet.Subnets[0].Id

# Enable encryption at host by specifying EncryptionAtHost parameter

$VMSS = New-AzVmssConfig -Location $LocationName -SkuCapacity 2 -SkuName $VMSize -UpgradePolicyMode
'Automatic' -EncryptionAtHost

$VMSS = Add-AzVmssNetworkInterfaceConfiguration -Name "myVMSSNetworkConfig" -VirtualMachineScaleSet $VMSS -
Primary $true -IpConfiguration $ipConfig

$VMSS = Set-AzVmssStorageProfile $VMSS -OsDiskCreateOption "FromImage" -ImageReferenceOffer 'WindowsServer'
-ImageReferenceSku '2012-R2-Datacenter' -ImageReferenceVersion latest -ImageReferencePublisher
'MicrosoftWindowsServer'

$VMSS = Set-AzVmssOsProfile $VMSS -ComputerNamePrefix $ComputerNamePrefix -AdminUsername $VMLocalAdminUser -
AdminPassword $VMLocalAdminSecurePassword

$VMSS = Add-AzVmssDataDisk -VirtualMachineScaleSet $VMSS -CreateOption Empty -Lun 1 -DiskSizeGB 128 -
StorageAccountType Premium_LRS

$Credential = New-Object System.Management.Automation.PSCredential ($VMLocalAdminUser,
$VMLocalAdminSecurePassword);

New-AzVmss -VirtualMachineScaleSet $VMSS -ResourceGroupName $ResourceGroupName -VMSScaleSetName
$VMSScaleSetName
```

### Update a virtual machine scale set to enable encryption at host.

```
$ResourceGroupName = "yourResourceGroupName"
$VMSScaleSetName = "yourVMSSName"

$VMSS = Get-AzVmss -ResourceGroupName $ResourceGroupName -Name $VMSScaleSetName

Update-AzVmss -VirtualMachineScaleSet $VMSS -Name $VMSScaleSetName -ResourceGroupName $ResourceGroupName -
EncryptionAtHost $true
```

### Check the status of encryption at host for a virtual machine scale set

```
$ResourceGroupName = "yourResourceGroupName"
$VMSScaleSetName = "yourVMSSName"

$VMSS = Get-AzVmss -ResourceGroupName $ResourceGroupName -Name $VMSScaleSetName

$VMSS.VirtualMachineProfile.SecurityProfile.EncryptionAtHost
```

### Update a virtual machine scale set to disable encryption at host.

You can disable encryption at host on your virtual machine scale set but, this will only affect VMs created after you disable encryption at host. For existing VMs, you must deallocate the VM, [disable encryption at host on that individual VM](#), then reallocate the VM.

```
$ResourceGroupName = "yourResourceGroupName"
$VMSScaleSetName = "yourVMSSName"

$VMSS = Get-AzVmss -ResourceGroupName $ResourceGroupName -Name $VMSScaleSetName

Update-AzVmss -VirtualMachineScaleSet $VMSS -Name $VMSScaleSetName -ResourceGroupName $ResourceGroupName -EncryptionAtHost $false
```

## Finding supported VM sizes

Legacy VM Sizes are not supported. You can find the list of supported VM sizes by either:

Calling the [Resource Skus API](#) and checking that the `EncryptionAtHostSupported` capability is set to `True`.

```
{
    "resourceType": "virtualMachines",
    "name": "Standard_DS1_v2",
    "tier": "Standard",
    "size": "DS1_v2",
    "family": "standardDSv2Family",
    "locations": [
        "CentralUSEUAP"
    ],
    "capabilities": [
        {
            "name": "EncryptionAtHostSupported",
            "value": "True"
        }
    ]
}
```

Or, calling the [Get-AzComputeResourceSku](#) PowerShell cmdlet.

```
$vmSizes=Get-AzComputeResourceSku | where{$_. ResourceType -eq 'virtualMachines' -and  
$_.Locations.Contains('CentralUSEUAP')}  
  
foreach($vmSize in $vmSizes)  
{  
    foreach($capability in $vmSize.capabilities)  
    {  
        if($capability.Name -eq 'EncryptionAtHostSupported' -and $capability.Value -eq 'true')  
        {  
            $vmSize  
  
        }  
    }  
}
```

## Next steps

Now that you've created and configured these resources, you can use them to secure your managed disks. The following link contains example scripts, each with a respective scenario, that you can use to secure your managed disks.

[Azure Resource Manager template samples](#)

# Use the Azure CLI to enable end-to-end encryption using encryption at host

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you enable encryption at host, data stored on the VM host is encrypted at rest and flows encrypted to the Storage service. For conceptual information on encryption at host, as well as other managed disk encryption types, see [Encryption at host - End-to-end encryption for your VM data](#).

## Restrictions

- Doesn't support ultra disks.
- Cannot be enabled if Azure Disk Encryption (guest-VM encryption using bitlocker/DM-Crypt) is enabled on your VMs/virtual machine scale sets.
- Azure Disk Encryption cannot be enabled on disks that have encryption at host enabled.
- The encryption can be enabled on existing virtual machine scale set. However, only new VMs created after enabling the encryption are automatically encrypted.
- Existing VMs must be deallocated and reallocated in order to be encrypted.
- Supports ephemeral OS disks but only with platform-managed keys.

## Supported VM sizes

The complete list of supported VM sizes can be pulled programmatically. To learn how to retrieve them programmatically, see the [Finding supported VM sizes](#) section. Upgrading the VM size will result in validation to check if the new VM size supports the EncryptionAtHost feature.

## Prerequisites

You must enable the feature for your subscription before you use the EncryptionAtHost property for your VM/VMSS. Use the following steps to enable the feature for your subscription:

- Execute the following command to register the feature for your subscription

```
az feature register --namespace Microsoft.Compute --name EncryptionAtHost
```

- Check that the registration state is **Registered** (takes a few minutes) using the command below before trying out the feature.

```
az feature show --namespace Microsoft.Compute --name EncryptionAtHost
```

## Create resources

Once the feature is enabled, you'll need to set up a DiskEncryptionSet and either an [Azure Key Vault](#) or an [Azure Key Vault Managed HSM](#).

### Azure Key Vault

- Install the latest [Azure CLI](#) and log to an Azure account in with `az login`.
- Create an Azure Key Vault and encryption key.

When creating the Key Vault, you must enable purge protection. Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

#### IMPORTANT

Don't camel case the region, if you do so, you may experience problems when assigning additional disks to the resource in the Azure portal.

```
subscriptionId=yourSubscriptionID
rgName=yourResourceGroupName
location=westcentralus
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName
diskName=yourDiskName

az account set --subscription $subscriptionId

az keyvault create -n $keyVaultName \
-g $rgName \
-l $location \
--enable-purge-protection true

az keyvault key create --vault-name $keyVaultName \
-n $keyName \
--protection software
```

- Create a DiskEncryptionSet. You can set enable-auto-key-rotation equal to true to enable automatic rotation of the key. When you enable automatic rotation, the system will automatically update all managed disks, snapshots, and images referencing the disk encryption set to use the new version of the key within one hour.

```
keyVaultKeyUrl=$(az keyvault key show --vault-name $keyVaultName --name $keyName --query [key.kid] -o tsv)

az disk-encryption-set create -n $diskEncryptionSetName
-l $location \
-g $rgName \
--key-url $keyVaultKeyUrl \
--enable-auto-key-rotation false
```

- Grant the DiskEncryptionSet resource access to the key vault.

#### NOTE

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
desIdentity=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [identity.principalId] -o tsv)

az keyvault set-policy -n $keyVaultName \
-g $rgName \
--object-id $desIdentity \
--key-permissions wrapkey unwrapkey get
```

## Azure Key Vault Managed HSM

Alternatively, you can use a Managed HSM to handle your keys.

To do this, you must complete the following prerequisites:

- Install the latest [Azure CLI](#) and log in to an Azure account in with `az login`.
- [Create and configure a managed HSM](#).
- [Assign permissions to a user, so they can manage your Managed HSM](#).

### Configuration

Once you've created a Managed HSM and added permissions, enable purge protection and create an encryption key.

```
subscriptionId=yourSubscriptionID
rgName=yourResourceGroupName
location=westcentralus
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName
diskName=yourDiskName

az account set --subscription $subscriptionId

az keyvault update-hsm --subscription $subscriptionId -g $rgName --hsm-name $keyVaultName --enable-purge-protection true

az keyvault key create --hsm-name $keyVaultName --name $keyName --ops wrapKey unwrapKey --kty RSA-HSM --size 2048
```

Then, create a DiskEncryptionSet.

```
keyVaultKeyUrl=$(az keyvault key show --vault-name $keyVaultName --name $keyName --query [key.kid] -o tsv)

az disk-encryption-set create -n $diskEncryptionSetName
-l $location \
-g $rgName \
--key-url $keyVaultKeyUrl \
--enable-auto-key-rotation false
```

Finally, grant the DiskEncryptionSet access to the Managed HSM.

```
desIdentity=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [identity.principalId] -o tsv)

az keyvault role assignment create --hsm-name $keyVaultName --role "Managed HSM Crypto Service Encryption User" --assignee $desIdentity --scope /keys
```

## Examples

### Create a VM with encryption at host enabled with customer-managed keys.

Create a VM with managed disks using the resource URI of the DiskEncryptionSet created earlier to encrypt cache of OS and data disks with customer-managed keys. The temp disks are encrypted with platform-managed keys.

```
rgName=yourRGName
vmName=yourVMName
location=eastus
vmSize=Standard_DS2_v2
image=UbuntuLTS
diskEncryptionSetName=yourDiskEncryptionSetName

diskEncryptionSetId=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [id] -o tsv)

az vm create -g $rgName \
-n $vmName \
-l $location \
--encryption-at-host \
--image $image \
--size $vmSize \
--generate-ssh-keys \
--os-disk-encryption-set $diskEncryptionSetId \
--data-disk-sizes-gb 128 128 \
--data-disk-encryption-sets $diskEncryptionSetId $diskEncryptionSetId
```

### Create a VM with encryption at host enabled with platform-managed keys.

Create a VM with encryption at host enabled to encrypt cache of OS/data disks and temp disks with platform-managed keys.

```
rgName=yourRGName
vmName=yourVMName
location=eastus
vmSize=Standard_DS2_v2
image=UbuntuLTS

az vm create -g $rgName \
-n $vmName \
-l $location \
--encryption-at-host \
--image $image \
--size $vmSize \
--generate-ssh-keys \
--data-disk-sizes-gb 128 128 \
```

### Update a VM to enable encryption at host.

```
rgName=yourRGName
vmName=yourVMName

az vm update -n $vmName \
-g $rgName \
--set securityProfile.encryptionAtHost=true
```

### Check the status of encryption at host for a VM

```
rgName=yourRGName
vmName=yourVMName

az vm show -n $vmName \
-g $rgName \
--query [securityProfile.encryptionAtHost] -o tsv
```

### Update a VM to disable encryption at host.

You must deallocate your VM before you can disable encryption at host.

```
rgName=yourRGName  
vmName=yourVMName  
  
az vm update -n $vmName \  
-g $rgName \  
--set securityProfile.encryptionAtHost=false
```

#### Create a virtual machine scale set with encryption at host enabled with customer-managed keys.

Create a virtual machine scale set with managed disks using the resource URI of the DiskEncryptionSet created earlier to encrypt cache of OS and data disks with customer-managed keys. The temp disks are encrypted with platform-managed keys.

```
rgName=yourRGName  
vmssName=yourVMSSName  
location=westus2  
vmSize=Standard_DS3_V2  
image=UbuntuLTS  
diskEncryptionSetName=yourDiskEncryptionSetName  
  
diskEncryptionSetId=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query [id] -o tsv)  
  
az vmss create -g $rgName \  
-n $vmssName \  
--encryption-at-host \  
--image UbuntuLTS \  
--upgrade-policy automatic \  
--admin-username azureuser \  
--generate-ssh-keys \  
--os-disk-encryption-set $diskEncryptionSetId \  
--data-disk-sizes-gb 64 128 \  
--data-disk-encryption-sets $diskEncryptionSetId $diskEncryptionSetId
```

#### Create a virtual machine scale set with encryption at host enabled with platform-managed keys.

Create a virtual machine scale set with encryption at host enabled to encrypt cache of OS/data disks and temp disks with platform-managed keys.

```
rgName=yourRGName  
vmssName=yourVMSSName  
location=westus2  
vmSize=Standard_DS3_V2  
image=UbuntuLTS  
  
az vmss create -g $rgName \  
-n $vmssName \  
--encryption-at-host \  
--image UbuntuLTS \  
--upgrade-policy automatic \  
--admin-username azureuser \  
--generate-ssh-keys \  
--data-disk-sizes-gb 64 128 \  
--data-disk-encryption-sets $diskEncryptionSetId $diskEncryptionSetId
```

#### Update a virtual machine scale set to enable encryption at host.

```
rgName=yourRGName  
vmssName=yourVMSSName  
  
az vmss update -n $vmssName \  
-g $rgName \  
--set virtualMachineProfile.securityProfile.encryptionAtHost=true
```

## Check the status of encryption at host for a virtual machine scale set

```
rgName=yourRGName  
vmssName=yourVMName  
  
az vmss show -n $vmssName \  
-g $rgName \  
--query [virtualMachineProfile.securityProfile.encryptionAtHost] -o tsv
```

## Update a virtual machine scale set to disable encryption at host.

You can disable encryption at host on your virtual machine scale set but, this will only affect VMs created after you disable encryption at host. For existing VMs, you must deallocate the VM, [disable encryption at host on that individual VM](#), then reallocate the VM.

```
rgName=yourRGName  
vmssName=yourVMName  
  
az vmss update -n $vmssName \  
-g $rgName \  
--set virtualMachineProfile.securityProfile.encryptionAtHost=false
```

## Finding supported VM sizes

Legacy VM Sizes are not supported. You can find the list of supported VM sizes by either:

Calling the [Resource Skus API](#) and checking that the `EncryptionAtHostSupported` capability is set to `True`.

```
{  
    "resourceType": "virtualMachines",  
    "name": "Standard_DS1_v2",  
    "tier": "Standard",  
    "size": "DS1_v2",  
    "family": "standardDSv2Family",  
    "locations": [  
        "CentralUSEUAP"  
    ],  
    "capabilities": [  
        {  
            "name": "EncryptionAtHostSupported",  
            "value": "True"  
        }  
    ]  
}
```

Or, calling the [Get-AzComputeResourceSku](#) PowerShell cmdlet.

```
$vmSizes=Get-AzComputeResourceSku | where{$_. ResourceType -eq 'virtualMachines' -and  
$_.Locations.Contains('CentralUSEUAP')}  
  
foreach($vmSize in $vmSizes)  
{  
    foreach($capability in $vmSize.capabilities)  
    {  
        if($capability.Name -eq 'EncryptionAtHostSupported' -and $capability.Value -eq 'true')  
        {  
            $vmSize  
  
        }  
    }  
}
```

## Next steps

Now that you've created and configured these resources, you can use them to secure your managed disks. The following link contains example scripts, each with a respective scenario, that you can use to secure your managed disks.

[Azure Resource Manager template samples](#)

# Use the Azure portal to enable double encryption at rest for managed disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓

Azure Disk Storage supports double encryption at rest for managed disks. For conceptual information on double encryption at rest, as well as other managed disk encryption types, see the [Double encryption at rest](#) section of our disk encryption article.

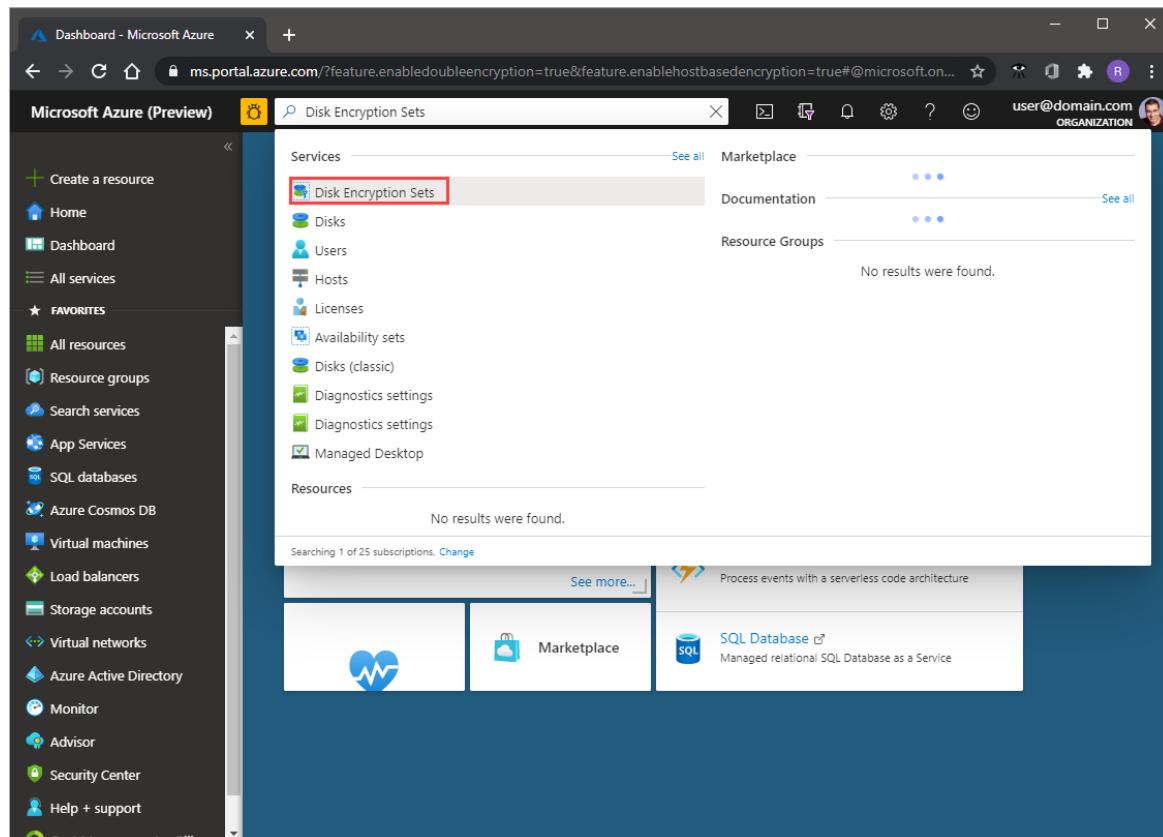
## Getting started

1. Sign in to the [Azure portal](#).

### IMPORTANT

You must use the [provided link](#) to access the Azure portal. Double encryption at rest is not currently visible in the public Azure portal without using the link.

2. Search for and select **Disk Encryption Sets**.



The screenshot shows the Microsoft Azure (Preview) portal interface. The left sidebar contains a list of services including Create a resource, Home, Dashboard, All services, Favorites, All resources, Resource groups, Search services, App Services, SQL databases, Azure Cosmos DB, Virtual machines, Load balancers, Storage accounts, Virtual networks, Azure Active Directory, Monitor, Advisor, Security Center, Help + support, and Cost Management + Bill. The main area has a search bar at the top with the text 'Disk Encryption Sets'. Below the search bar, there are two sections: 'Services' and 'Resources'. In the 'Services' section, 'Disk Encryption Sets' is highlighted with a red box. Other items listed include Disks, Users, Hosts, Licenses, Availability sets, Disks (classic), Diagnostics settings, Diagnostics settings, and Managed Desktop. In the 'Resources' section, it says 'No results were found.' At the bottom, there are cards for Process events with a serverless code architecture, Marketplace, and SQL Database.

3. Select **+ Add**.



The screenshot shows the 'Disk Encryption Sets' blade. At the top, it says 'Disk Encryption Sets' and 'Microsoft'. Below that are three buttons: '+ Add' (highlighted with a red box), 'Manage view', and 'Refresh'. The '+ Add' button has a blue plus icon and the word 'Add'.

4. Select one of the supported regions.
5. For **Encryption type**, select **Double encryption with platform-managed and customer-managed keys**.

**NOTE**

Once you create a disk encryption set with a particular encryption type, it cannot be changed. If you want to use a different encryption type, you must create a new disk encryption set.

6. Fill in the remaining info.

## Create a disk encryption set

Basics Tags Review + create

Disk encryption sets allow you to manage encryption keys using server-side encryption for Standard HDD, Standard SSD, and Premium SSD managed disks. It will give you control of the encryption keys to meet your security and compliance needs in a few clicks. [Learn more about disk encryption sets](#).

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

My Example Subscription

Resource group \* ⓘ

myResourceGroup

[Create new](#)

### Instance details

Disk encryption set name \*

my-disk-encryption

Region \* ⓘ

(US) West US 2

Encryption type \* ⓘ

Double encryption with platform-managed and customer-managed keys

Key vault and key \*

Key vault: theexamplekeyvault  
Key: mykey  
Version: 3d033002b6d84006ad1db570dbc72f13  
[Click to select a key](#)

7. Select an Azure Key Vault and key, or create a new one if necessary.

**NOTE**

If you create a Key Vault instance, you must enable soft delete and purge protection. These settings are mandatory when using a Key Vault for encrypting managed disks, and protect you from losing data due to accidental deletion.

## Select key from Azure Key Vault

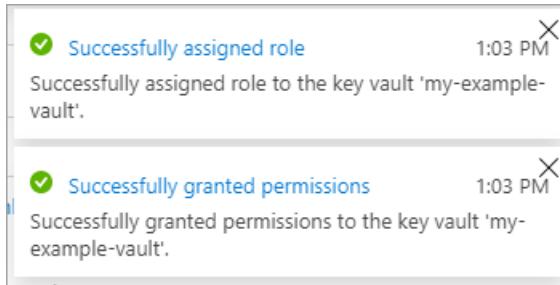
Key vault *	theexamplekeyvault
	<a href="#">Create new</a>
Key *	mykey
	<a href="#">Create new</a>
Version * ⓘ	
	<a href="#">Create new</a>

8. Select **Create**.

9. Navigate to the disk encryption set you created, and select the error that is displayed. This will configure your disk encryption set to work.

 To associate a disk, image, or snapshot with this disk encryption set, you must grant permissions to the key vault → theexamplekeyvault.

A notification should pop up and succeed. Doing this will allow you to use the disk encryption set with your key vault.



10. Navigate to your disk.

11. Select **Encryption**.

12. For **Encryption type**, select **Double encryption with platform-managed and customer-managed keys**.

13. Select your disk encryption set.

14. select **Save**.

### data-detach | Encryption

Disk

Search (Ctrl+/) <> Save Discard

Encryption type \*

Double encryption with platform-managed and customer-managed keys

Once a customer-managed key is used, you can't change the selection back to a platform-managed key. [Learn more about disk encryption](#).

Disk encryption set \* ⓘ

example-disk-enc-set

Encryption

Configuration

Disk Export

You have now enabled double encryption at rest on your managed disk.

## Next steps

- [Azure PowerShell - Enable customer-managed keys with server-side encryption - managed disks](#)
- [Azure Resource Manager template samples](#)
- [Enable customer-managed keys with server-side encryption - Examples](#)

# Use the Azure PowerShell module to enable double encryption at rest for managed disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Azure Disk Storage supports double encryption at rest for managed disks. For conceptual information on double encryption at rest, as well as other managed disk encryption types, see the [Double encryption at rest](#) section of our disk encryption article.

## Prerequisites

Install the latest [Azure PowerShell version](#), and sign in to an Azure account using [Connect-AzAccount](#).

## Getting started

1. Create an instance of Azure Key Vault and encryption key.

When creating the Key Vault instance, you must enable soft delete and purge protection. Soft delete ensures that the Key Vault holds a deleted key for a given retention period (90 day default). Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

```
$ResourceGroupName="yourResourceGroupName"
$LocationName="westus2"
$keyVaultName="yourKeyVaultName"
$keyName="yourKeyName"
$keyDestination="Software"
$diskEncryptionSetName="yourDiskEncryptionSetName"

$keyVault = New-AzKeyVault -Name $keyVaultName -ResourceGroupName $ResourceGroupName -Location
$LocationName -EnableSoftDelete -EnablePurgeProtection

$key = Add-AzKeyVaultKey -VaultName $keyVaultName -Name $keyName -Destination $keyDestination
```

2. Create a DiskEncryptionSet with encryptionType set as EncryptionAtRestWithPlatformAndCustomerKeys.

Use API version **2020-05-01** in the Azure Resource Manager (ARM) template.

```
New-AzResourceGroupDeployment -ResourceGroupName $ResourceGroupName ` 
-TemplateUri "https://raw.githubusercontent.com/Azure-Samples/managed-disks-powershell-getting-` 
started/master/DoubleEncryption/CreateDiskEncryptionSetForDoubleEncryption.json" ` 
-diskEncryptionSetName $diskEncryptionSetName ` 
-keyVaultId $keyVault.ResourceId ` 
-keyVaultKeyUrl $key.Key.Kid ` 
-encryptionType "EncryptionAtRestWithPlatformAndCustomerKeys" ` 
-region $LocationName
```

3. Grant the DiskEncryptionSet resource access to the key vault.

#### **NOTE**

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
$des=Get-AzDiskEncryptionSet -name $diskEncryptionSetName -ResourceGroupName $ResourceGroupName  
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ObjectId $des.Identity.PrincipalId -  
PermissionsToKeys wrapkey,unwrapkey,get
```

## Next steps

Now that you've created and configured these resources, you can use them to secure your managed disks. The following links contain example scripts, each with a respective scenario, that you can use to secure your managed disks.

- [Azure PowerShell - Enable customer-managed keys with server-side encryption - managed disks](#)
- [Azure Resource Manager template samples](#)

# Use the Azure CLI to enable double encryption at rest for managed disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure Disk Storage supports double encryption at rest for managed disks. For conceptual information on double encryption at rest, as well as other managed disk encryption types, see the [Double encryption at rest](#) section of our disk encryption article.

## Prerequisites

Install the latest [Azure CLI](#) and log in to an Azure account with [az login](#).

## Getting started

1. Create an instance of Azure Key Vault and encryption key.

When creating the Key Vault instance, you must enable soft delete and purge protection. Soft delete ensures that the Key Vault holds a deleted key for a given retention period (90 day default). Purge protection ensures that a deleted key cannot be permanently deleted until the retention period lapses. These settings protect you from losing data due to accidental deletion. These settings are mandatory when using a Key Vault for encrypting managed disks.

```
subscriptionId=yourSubscriptionID
rgName=yourResourceGroupName
location=westcentralus
keyVaultName=yourKeyVaultName
keyName=yourKeyName
diskEncryptionSetName=yourDiskEncryptionSetName
diskName=yourDiskName

az account set --subscription $subscriptionId

az keyvault create -n $keyVaultName -g $rgName -l $location --enable-purge-protection true --enable-soft-delete true

az keyvault key create --vault-name $keyVaultName -n $keyName --protection software
```

2. Create a DiskEncryptionSet with encryptionType set as EncryptionAtRestWithPlatformAndCustomerKeys.

Use API version **2020-05-01** in the Azure Resource Manager (ARM) template.

```
az deployment group create -g $rgName \
--template-uri "https://raw.githubusercontent.com/Azure-Samples/managed-disks-powershell-getting-started/master/DoubleEncryption/CreateDiskEncryptionSetForDoubleEncryption.json" \
--parameters "diskEncryptionSetName=$diskEncryptionSetName"
"encryptionType=EncryptionAtRestWithPlatformAndCustomerKeys" "keyVaultId=$keyVaultId"
"keyVaultKeyUrl=$keyVaultKeyUrl" "region=$location"
```

3. Grant the DiskEncryptionSet resource access to the key vault.

#### NOTE

It may take few minutes for Azure to create the identity of your DiskEncryptionSet in your Azure Active Directory. If you get an error like "Cannot find the Active Directory object" when running the following command, wait a few minutes and try again.

```
desIdentity=$(az disk-encryption-set show -n $diskEncryptionSetName -g $rgName --query  
[identity.principalId] -o tsv)  
  
az keyvault set-policy -n $keyVaultName -g $rgName --object-id $desIdentity --key-permissions wrapkey  
unwrapkey get
```

## Next steps

Now that you've created and configured these resources, you can use them to secure your managed disks. The following links contain example scripts, each with a respective scenario, that you can use to secure your managed disks.

- [Azure Resource Manager template samples](#)
- [Enable customer-managed keys with server-side encryption - Examples](#)

# Azure Disk Encryption for Linux VMs

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure Disk Encryption helps protect and safeguard your data to meet your organizational security and compliance commitments. It uses the [DM-Crypt](#) feature of Linux to provide volume encryption for the OS and data disks of Azure virtual machines (VMs), and is integrated with [Azure Key Vault](#) to help you control and manage the disk encryption keys and secrets.

Azure Disk Encryption is zone resilient, the same way as Virtual Machines. For details, see [Azure Services that support Availability Zones](#).

If you use [Microsoft Defender for Cloud](#), you're alerted if you have VMs that aren't encrypted. The alerts show as High Severity and the recommendation is to encrypt these VMs.

VIRTUAL MACHINES RECOMMENDATIONS		TOTAL			
Missing disk encryption	2 of 2 VMs	<div style="width: 100%; background-color: red; height: 10px;"></div>			
Virtual machines					
NAME	ONBOARDING	SYSTEM UPDATES	ANTIMALWARE	BASELINE	DISK ENCRYPTION
ASC-VM1	✓	✓	✓	✓	!
ASC-VM2	✓	✓	✓	✓	!

## WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue to use this option to encrypt your VM. See [Azure Disk Encryption with Azure AD \(previous release\)](#) for details.
- Certain recommendations might increase data, network, or compute resource usage, resulting in additional license or subscription costs. You must have a valid active Azure subscription to create resources in Azure in the supported regions.

You can learn the fundamentals of Azure Disk Encryption for Linux in just a few minutes with the [Create and encrypt a Linux VM with Azure CLI quickstart](#) or the [Create and encrypt a Linux VM with Azure PowerShell quickstart](#).

## Supported VMs and operating systems

### Supported VMs

Linux VMs are available in a [range of sizes](#). Azure Disk Encryption is supported on Generation 1 and Generation 2 VMs. Azure Disk Encryption is also available for VMs with premium storage.

See [Azure VM sizes with no local temporary disk](#).

Azure Disk Encryption is also not available on [Basic, A-series VMs](#), or on virtual machines that do not meet these minimum memory requirements:

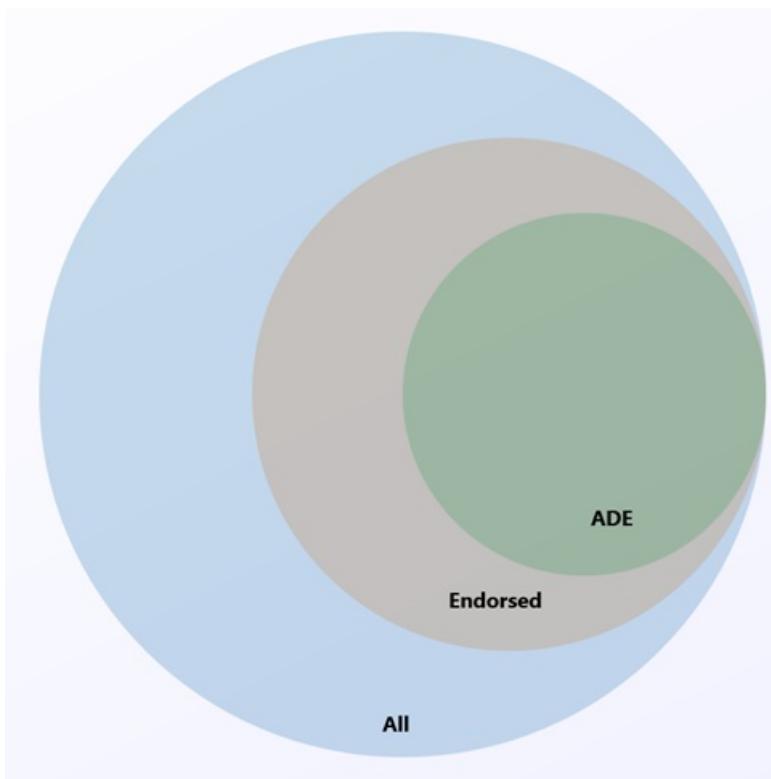
VIRTUAL MACHINE	MINIMUM MEMORY REQUIREMENT
Linux VMs when only encrypting data volumes	2 GB
Linux VMs when encrypting both data and OS volumes, and where the root (/) file system usage is 4GB or less	8 GB
Linux VMs when encrypting both data and OS volumes, and where the root (/) file system usage is greater than 4GB	The root file system usage * 2. For instance, a 16 GB of root file system usage requires at least 32GB of RAM

Once the OS disk encryption process is complete on Linux virtual machines, the VM can be configured to run with less memory.

For more exceptions, see [Azure Disk Encryption: Unsupported scenarios](#).

### Supported operating systems

Azure Disk Encryption is supported on a subset of the [Azure-endorsed Linux distributions](#), which is itself a subset of all Linux server possible distributions.



Linux server distributions that are not endorsed by Azure do not support Azure Disk Encryption; of those that are endorsed, only the following distributions and versions support Azure Disk Encryption:

PUBLISHER	OFFER	SKU	URN	VOLUME TYPE SUPPORTED FOR ENCRYPTION
Canonical	Ubuntu	20.04-LTS	Canonical:0001-com-ubuntu-server-focal:20_04-lts:latest	OS and data disk
Canonical	Ubuntu	20.04-DAILY-LTS	Canonical:0001-com-ubuntu-server-focal-daily:20_04-daily-lts:latest	OS and data disk

PUBLISHER	OFFER	SKU	URN	VOLUME TYPE SUPPORTED FOR ENCRYPTION
Canonical	Ubuntu	20.04-LTS Gen2	Canonical:0001-com-ubuntu-server-focal:20_04-lts-gen2:latest	OS and data disk
Canonical	Ubuntu	20.04-DAILY-LTS Gen2	Canonical:0001-com-ubuntu-server-focal-daily:20_04-daily-lts-gen2:latest	OS and data disk
Canonical	Ubuntu	18.04-LTS	Canonical:UbuntuServer:18.04-LTS:latest	OS and data disk
Canonical	Ubuntu 18.04	18.04-DAILY-LTS	Canonical:UbuntuServer:18.04-DAILY-LTS:latest	OS and data disk
Canonical	Ubuntu 16.04	16.04-DAILY-LTS	Canonical:UbuntuServer:16.04-DAILY-LTS:latest	OS and data disk
Canonical	Ubuntu 14.04.5 with Azure tuned kernel updated to 4.15 or later	14.04.5-LTS	Canonical:UbuntuServer:14.04.5-LTS:latest	OS and data disk
Canonical	Ubuntu 14.04.5 with Azure tuned kernel updated to 4.15 or later	14.04.5-DAILY-LTS	Canonical:UbuntuServer:14.04.5-DAILY-LTS:latest	OS and data disk
Oracle	Oracle Linux 8.5 (Public Preview)	8.5	Oracle:Oracle-Linux:ol85-lvm:latest	OS and data disk (see note below)
Oracle	Oracle Linux 8.5 Gen 2 (Public Preview)	8.5	Oracle:Oracle-Linux:ol85-lvm-gen2:latest	OS and data disk (see note below)
RedHat	RHEL 8.6	8.6	RedHat:RHEL:8_6:latest	OS and data disk (see note below)
RedHat	RHEL 8.6 Gen 2	8.5	RedHat:RHEL:86-gen2:latest	OS and data disk (see note below)
RedHat	RHEL 8.5	8.5	RedHat:RHEL:8_5:latest	OS and data disk (see note below)
RedHat	RHEL 8.5 Gen 2	8.5	RedHat:RHEL:85-gen2:latest	OS and data disk (see note below)
RedHat	RHEL 8.4	8.4	RedHat:RHEL:8.4:latest	OS and data disk (see note below)

PUBLISHER	OFFER	SKU	URN	VOLUME TYPE SUPPORTED FOR ENCRYPTION
RedHat	RHEL 8.3	8.3	RedHat:RHEL:8.3:latest	OS and data disk (see note below)
RedHat	RHEL 8-LVM	8-LVM	RedHat:RHEL:8-LVM:8.2.20200905	OS and data disk (see note below)
RedHat	RHEL 8.2	8.2	RedHat:RHEL:8.2:latest	OS and data disk (see note below)
RedHat	RHEL 8.1	8.1	RedHat:RHEL:8.1:latest	OS and data disk (see note below)
RedHat	RHEL 7-LVM	7-LVM	RedHat:RHEL:7-LVM:7.9.2020111202	OS and data disk (see note below)
RedHat	RHEL 7.9	7_9	RedHat:RHEL:7_9:latest	OS and data disk (see note below)
RedHat	RHEL 7.8	7.8	RedHat:RHEL:7.8:latest	OS and data disk (see note below)
RedHat	RHEL 7.7	7.7	RedHat:RHEL:7.7:latest	OS and data disk (see note below)
RedHat	RHEL 7.6	7.6	RedHat:RHEL:7.6:latest	OS and data disk (see note below)
RedHat	RHEL 7.5	7.5	RedHat:RHEL:7.5:latest	OS and data disk (see note below)
RedHat	RHEL 7.4	7.4	RedHat:RHEL:7.4:latest	OS and data disk (see note below)
RedHat	RHEL 7.3	7.3	RedHat:RHEL:7.3:latest	OS and data disk (see note below)
RedHat	RHEL 7.2	7.2	RedHat:RHEL:7.2:latest	OS and data disk (see note below)
RedHat	RHEL 6.8	6.8	RedHat:RHEL:6.8:latest	Data disk (see note below)
RedHat	RHEL 6.7	6.7	RedHat:RHEL:6.7:latest	Data disk (see note below)
OpenLogic	CentOS 8-LVM	8-LVM	OpenLogic:CentOS-LVM:8-LVM:latest	OS and data disk
OpenLogic	CentOS 8.4	8_4	OpenLogic:CentOS:8_4:latest	OS and data disk

PUBLISHER	OFFER	SKU	URN	VOLUME TYPE SUPPORTED FOR ENCRYPTION
OpenLogic	CentOS 8.3	8_3	OpenLogic:CentOS:8_3:latest	OS and data disk
OpenLogic	CentOS 8.2	8_2	OpenLogic:CentOS:8_2:latest	OS and data disk
OpenLogic	CentOS 8.1	8_1	OpenLogic:CentOS:8_1:latest	OS and data disk
OpenLogic	CentOS 7-LVM	7-LVM	OpenLogic:CentOS-LVM:7-LVM:7.9.2021020400	OS and data disk
OpenLogic	CentOS 7.9	7_9	OpenLogic:CentOS:7_9:latest	OS and data disk
OpenLogic	CentOS 7.8	7_8	OpenLogic:CentOS:7_8:latest	OS and data disk
OpenLogic	CentOS 7.7	7.7	OpenLogic:CentOS:7.7:latest	OS and data disk
OpenLogic	CentOS 7.6	7.6	OpenLogic:CentOS:7.6:latest	OS and data disk
OpenLogic	CentOS 7.5	7.5	OpenLogic:CentOS:7.5:latest	OS and data disk
OpenLogic	CentOS 7.4	7.4	OpenLogic:CentOS:7.4:latest	OS and data disk
OpenLogic	CentOS 7.3	7.3	OpenLogic:CentOS:7.3:latest	OS and data disk
OpenLogic	CentOS 7.2n	7.2n	OpenLogic:CentOS:7.2n:latest	OS and data disk
OpenLogic	CentOS 7.1	7.1	OpenLogic:CentOS:7.1:latest	Data disk only
OpenLogic	CentOS 7.0	7.0	OpenLogic:CentOS:7.0:latest	Data disk only
OpenLogic	CentOS 6.8	6.8	OpenLogic:CentOS:6.8:latest	Data disk only
SUSE	openSUSE 42.3	42.3	SUSE:openSUSE-Leap:42.3:latest	Data disk only
SUSE	SLES 12-SP4	12-SP4	SUSE:SLES:12-SP4:latest	Data disk only

PUBLISHER	OFFER	SKU	URN	VOLUME TYPE SUPPORTED FOR ENCRYPTION
SUSE	SLES HPC 12-SP3	12-SP3	SUSE:SLES-HPC:12-SP3:latest	Data disk only

#### NOTE

The new Azure Disk Encryption implementation is supported for RHEL OS and data disk for RHEL7 Pay-As-You-Go images.

ADE is also supported for RHEL Bring-Your-Own-Subscription Gold Images, but only **after** the subscription has been registered . For more information, see [Red Hat Enterprise Linux Bring-Your-Own-Subscription Gold Images in Azure](#)

ADE support for a particular offer type does not extend beyond the end-of-life date provided by the publisher.

The legacy ADE solution (using AAD credentials) is not recommended for new VM's and is not compatible with RHEL versions later than RHEL 7.8.

## Additional VM requirements

Azure Disk Encryption requires the dm-crypt and vfat modules to be present on the system. Removing or disabling vfat from the default image will prevent the system from reading the key volume and obtaining the key needed to unlock the disks on subsequent reboots. System hardening steps that remove the vfat module from the system or enforce expanding the OS mountpoints/folders on data drives are not compatible with Azure Disk Encryption.

Before enabling encryption, the data disks to be encrypted must be properly listed in /etc/fstab. Use the "nofail" option when creating entries, and choose a persistent block device name (as device names in the "/dev/sdX" format may not be associated with the same disk across reboots, particularly after encryption; for more detail on this behavior, see: [Troubleshoot Linux VM device name changes](#)).

Make sure the /etc/fstab settings are configured properly for mounting. To configure these settings, run the mount -a command or reboot the VM and trigger the remount that way. Once that is complete, check the output of the lsblk command to verify that the drive is still mounted.

- If the /etc/fstab file doesn't mount the drive properly before enabling encryption, Azure Disk Encryption won't be able to mount it properly.
- The Azure Disk Encryption process will move the mount information out of /etc/fstab and into its own configuration file as part of the encryption process. Don't be alarmed to see the entry missing from /etc/fstab after data drive encryption completes.
- Before starting encryption, be sure to stop all services and processes that could be writing to mounted data disks and disable them, so that they do not restart automatically after a reboot. These could keep files open on these partitions, preventing the encryption procedure to remount them, causing failure of the encryption.
- After reboot, it will take time for the Azure Disk Encryption process to mount the newly encrypted disks. They won't be immediately available after a reboot. The process needs time to start, unlock, and then mount the encrypted drives before being available for other processes to access. This process may take more than a minute after reboot depending on the system characteristics.

Here is an example of the commands used to mount the data disks and create the necessary /etc/fstab entries:

```

UUID0=$(blkid -s UUID -o value /dev/sda1)
UUID1=$(blkid -s UUID -o value /dev/sda2)
mkdir /data0
mkdir /data1
echo "UUID=$UUID0 /data0 ext4 defaults,nofail 0 0" >>/etc/fstab
echo "UUID=$UUID1 /data1 ext4 defaults,nofail 0 0" >>/etc/fstab
mount -a

```

## Networking requirements

To enable the Azure Disk Encryption feature, the Linux VMs must meet the following network endpoint configuration requirements:

- To get a token to connect to your key vault, the Linux VM must be able to connect to an Azure Active Directory endpoint, [login.microsoftonline.com].
- To write the encryption keys to your key vault, the Linux VM must be able to connect to the key vault endpoint.
- The Linux VM must be able to connect to an Azure storage endpoint that hosts the Azure extension repository and an Azure storage account that hosts the VHD files.
- If your security policy limits access from Azure VMs to the Internet, you can resolve the preceding URI and configure a specific rule to allow outbound connectivity to the IPs. For more information, see [Azure Key Vault behind a firewall](#).

## Encryption key storage requirements

Azure Disk Encryption requires an Azure Key Vault to control and manage disk encryption keys and secrets. Your key vault and VMs must reside in the same Azure region and subscription.

For details, see [Creating and configuring a key vault for Azure Disk Encryption](#).

## Terminology

The following table defines some of the common terms used in Azure disk encryption documentation:

TERMINOLOGY	DEFINITION
Azure Key Vault	Key Vault is a cryptographic, key management service that's based on Federal Information Processing Standards (FIPS) validated hardware security modules. These standards help to safeguard your cryptographic keys and sensitive secrets. For more information, see the <a href="#">Azure Key Vault</a> documentation and <a href="#">Creating and configuring a key vault for Azure Disk Encryption</a> .
Azure CLI	The <a href="#">Azure CLI</a> is optimized for managing and administering Azure resources from the command line.
DM-Crypt	<a href="#">DM-Crypt</a> is the Linux-based, transparent disk-encryption subsystem that's used to enable disk encryption on Linux VMs.

TERMINOLOGY	DEFINITION
Key encryption key (KEK)	The asymmetric key (RSA 2048) that you can use to protect or wrap the secret. You can provide a hardware security module (HSM)-protected key or software-protected key. For more information, see the <a href="#">Azure Key Vault</a> documentation and <a href="#">Creating and configuring a key vault for Azure Disk Encryption</a> .
PowerShell cmdlets	For more information, see <a href="#">Azure PowerShell cmdlets</a> .

## Next steps

- [Quickstart - Create and encrypt a Linux VM with Azure CLI](#)
- [Quickstart - Create and encrypt a Linux VM with Azure PowerShell](#)
- [Azure Disk Encryption scenarios on Linux VMs](#)
- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)
- [Creating and configuring a key vault for Azure Disk Encryption](#)

# Quickstart: Create and encrypt a Linux VM with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The Azure CLI is used to create and manage Azure resources from the command line or in scripts. This quickstart shows you how to use the Azure CLI to create and encrypt a Linux virtual machine (VM).

If you don't have an Azure subscription, create a [free account](#) before you begin.

If you choose to install and use the Azure CLI locally, this quickstart requires that you are running the Azure CLI version 2.0.30 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

## Create a resource group

Create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name "myResourceGroup" --location "eastus"
```

## Create a virtual machine

Create a VM with `az vm create`. The following example creates a VM named *myVM*.

```
az vm create \
  --resource-group "myResourceGroup" \
  --name "myVM" \
  --image "Canonical:UbuntuServer:16.04-LTS:latest" \
  --size "Standard_D2S_V3" \
  --generate-ssh-keys
```

It takes a few minutes to create the VM and supporting resources. The following example output shows the VM create operation was successful.

```
{
  "fqdns": "",
  "id": "/subscriptions/<guid>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",
  "location": "eastus",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "52.174.34.95",
  "resourceGroup": "myResourceGroup"
}
```

## Create a Key Vault configured for encryption keys

Azure disk encryption stores its encryption key in an Azure Key Vault. Create a Key Vault with [az keyvault create](#). To enable the Key Vault to store encryption keys, use the --enabled-for-disk-encryption parameter.

#### IMPORTANT

Every key vault must have a name that is unique across Azure. In the examples below, replace <your-unique-keyvault-name> with the name you choose.

```
az keyvault create --name "<your-unique-keyvault-name>" --resource-group "myResourceGroup" --location "eastus" --enabled-for-disk-encryption
```

## Encrypt the virtual machine

Encrypt your VM with [az vm encryption](#), providing your unique Key Vault name to the --disk-encryption-keyvault parameter.

```
az vm encryption enable -g "MyResourceGroup" --name "myVM" --disk-encryption-keyvault "<your-unique-keyvault-name>"
```

After a moment the process will return, "The encryption request was accepted. Please use 'show' command to monitor the progress.". The "show" command is [az vm show](#).

```
az vm encryption show --name "myVM" -g "MyResourceGroup"
```

When encryption is enabled, you will see the following in the returned output:

```
"EncryptionOperation": "EnableEncryption"
```

## Clean up resources

When no longer needed, you can use the [az group delete](#) command to remove the resource group, VM, and Key Vault.

```
az group delete --name "myResourceGroup"
```

## Next steps

In this quickstart, you created a virtual machine, created a Key Vault that was enable for encryption keys, and encrypted the VM. Advance to the next article to learn more about more Azure Disk Encryption for Linux VMs.

[Azure Disk Encryption overview](#)

# Quickstart: Create and encrypt a Linux VM in Azure with Azure PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The Azure PowerShell module is used to create and manage Azure resources from the PowerShell command line or in scripts. This quickstart shows you how to use the Azure PowerShell module to create a Linux virtual machine (VM), create a Key Vault for the storage of encryption keys, and encrypt the VM. This quickstart uses the Ubuntu 16.04 LTS marketplace image from Canonical and a VM Standard\_D2S\_V3 size.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Create a resource group

Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which Azure resources are deployed and managed:

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

## Create a virtual machine

Create an Azure virtual machine with [New-AzVM](#), passing to it the VM configuration object you created above.

```
$cred = Get-Credential  
  
New-AzVM -Name MyVm -Credential $cred -ResourceGroupName MyResourceGroup -Image  
Canonical:UbuntuServer:18.04-LTS:latest -Size Standard_D2S_V3
```

It will take a few minutes for your VM to be deployed.

## Create a Key Vault configured for encryption keys

Azure disk encryption stores its encryption key in an Azure Key Vault. Create a Key Vault with [New-AzKeyvault](#). To enable the Key Vault to store encryption keys, use the `-EnabledForDiskEncryption` parameter.

### IMPORTANT

Every key vault must have a name that is unique across Azure. In the examples below, replace `<your-unique-keyvault-name>` with the name you choose.

```
New-AzKeyvault -name "<your-unique-keyvault-name>" -ResourceGroupName "myResourceGroup" -Location EastUS -  
EnabledForDiskEncryption
```

## Encrypt the virtual machine

Encrypt your VM with [Set-AzVmDiskEncryptionExtension](#).

Set-AzVmDiskEncryptionExtension requires some values from your Key Vault object. You can obtain these values by passing the unique name of your key vault to [Get-AzKeyvault](#).

```
$KeyVault = Get-AzKeyVault -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup"

Set-AzVmDiskEncryptionExtension -ResourceGroupName MyResourceGroup -VMName "MyVM" -DiskEncryptionKeyVaultUrl
$KeyVault.VaultUri -DiskEncryptionKeyId $KeyVault.ResourceId -SkipVmBackup -VolumeType All
```

After a few minutes the process will return the following:

RequestId	IsSuccessStatusCode	StatusCode	ReasonPhrase
-----	-----	-----	-----
True		OK	OK

You can verify the encryption process by running [Get-AzVmDiskEncryptionStatus](#).

```
Get-AzVmDiskEncryptionStatus -VMName MyVM -ResourceGroupName MyResourceGroup
```

When encryption is enabled, you will see the following in the returned output:

OsVolumeEncrypted	:	EncryptionInProgress
DataVolumesEncrypted	:	NotMounted
OsVolumeEncryptionSettings	:	Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage	:	OS disk encryption started

## Clean up resources

When no longer needed, you can use the [Remove-AzResourceGroup](#) cmdlet to remove the resource group, VM, and all related resources:

```
Remove-AzResourceGroup -Name "myResourceGroup"
```

## Next steps

In this quickstart, you created a virtual machine, created a Key Vault that was enable for encryption keys, and encrypted the VM. Advance to the next article to learn more about Azure Disk Encryption for Linux VMs.

[Azure Disk Encryption overview](#)

# Quickstart: Create and encrypt a virtual machine with the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure virtual machines (VMs) can be created through the Azure portal. The Azure portal is a browser-based user interface to create VMs and their associated resources. In this quickstart you will use the Azure portal to deploy a Linux virtual machine (VM) running Ubuntu 18.04 LTS, create a key vault for the storage of encryption keys, and encrypt the VM.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Sign in to Azure

Sign in to the [Azure portal](#).

## Create a virtual machine

1. Choose **Create a resource** in the upper left corner of the Azure portal.
2. In the New page, under Popular, select **Ubuntu Server 18.04 LTS**.
3. In the Basics tab, under Project details, verify sure the correct subscription is selected.
4. For "Resource Group", select **Create new**. Enter *myResourceGroup* as the name and select **Ok**.
5. For **Virtual machine name**, enter *MyVM*.
6. For **Region**, select *(US) East US*.
7. Make sure the **Size** is *Standard D2s v3*.
8. Under **Administrator account**, select *Password* as the **Authentication type**. Enter a user name and a password.

## Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Create a virtual machine that runs Linux or Windows. Select an image from Azure marketplace or use your own customized image. Complete the Basics tab then Review + create to provision a virtual machine with default parameters or review each tab for full customization. [Learn more ↗](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ	<input type="text" value="Free trial"/>
Resource group * ⓘ	<input type="text" value="(New) myResourceGroup"/> <span style="float: right;">▼</span> <a href="#">Create new</a>

### Instance details

Virtual machine name * ⓘ	<input type="text" value="myVM"/> <span style="float: right;">✓</span>
Region * ⓘ	<input type="text" value="(US) East US"/> <span style="float: right;">▼</span>
Availability options ⓘ	<input type="text" value="No infrastructure redundancy required"/> <span style="float: right;">▼</span>
Image * ⓘ	<input type="text" value="Ubuntu Server 18.04 LTS - Gen1"/> <span style="float: right;">▼</span> <a href="#">Browse all public and private images</a>

Azure Spot instance ⓘ  Yes  No

Size \* ⓘ  ▼

### Administrator account

Authentication type ⓘ  SSH public key  Password

Username \* ⓘ  ✓

Password \* ⓘ  ✓

Confirm password \* ⓘ  ✓

### WARNING

The "Disks" tab features an "Encryption Type" field under Disk options. This field is used to specify encryption options for [Managed Disks + CMK](#), not for Azure Disk Encryption.

To avoid confusion, we suggest you skip the *Disks* tab entirely while completing this tutorial.

9. Select the "Management" tab and verify that you have a Diagnostics Storage Account. If you have no storage accounts, select *Create New*, name your storage account *myStorageAccount*, and select "Ok"

The screenshot shows the Azure portal interface for creating a virtual machine. On the left, the 'Create a virtual machine' wizard is open, with the 'Management' tab selected (indicated by a red circle). The 'Networking' tab is also highlighted in blue. On the right, a separate 'Create storage account' dialog is displayed, also with the 'Management' tab selected. The 'Name' field contains 'adestorageaccount' with a green checkmark and '.core.windows.net'. The 'Account kind' dropdown is set to 'Storage (general purpose v1)'. Under 'Performance', 'Standard' is selected. Under 'Replication', 'Locally-redundant storage (LRS)' is selected. A red circle highlights the 'Create new' link under the 'Diagnostics storage account' section, which has a tooltip 'No existing storage accounts in current subscription'.

10. Click "Review + Create".
11. On the **Create a virtual machine** page, you can see the details about the VM you are about to create. When you are ready, select **Create**.

It will take a few minutes for your VM to be deployed. When the deployment is finished, move on to the next section.

## Encrypt the virtual machine

1. When the VM deployment is complete, select **Go to resource**.
2. On the left-hand sidebar, select **Disks**.
3. On the top bar, select **Additional Settings**.
4. Under **Encryption settings > Disks to encrypt**, select OS and data disks.

## Disk settings

myVM

### Ultra disk

Enable Ultra disk compatibility ⓘ

Yes

No

 Ultra disk is available only for Availability Zones in eastus. [Learn more ↗](#)

### Encryption settings

Azure Disk Encryption (ADE) provides volume encryption for the OS and data disks. [Learn more about Azure Disk Encryption.](#)

Disks to encrypt ⓘ

None

None

OS disk

Data disks

OS and data disks



5. Under **Encryption settings**, choose **Select a key vault and key for encryption**.

6. On the **Select key from Azure Key Vault** screen, select **Create New**.

Home > myVM - Disks > Encryption > Select key from Azure Key Vault

### Select key from Azure Key Vault

Key vault \*

Select the key vault.

Key

Select the key.

Version ⓘ

Select the version.

7. To the left of **Key vault** and **key**, select **Click to select a key**.

8. On the **Select key from Azure Key Vault**, under the **Key Vault** field, select **Create new**.

9. On the **Create key vault** screen, ensure that the Resource Group is *myResourceGroup*, and give your key vault a name. Every key vault across Azure must have an unique name.

10. On the **Access Policies** tab, check the **Azure Disk Encryption for volume encryption** box.

## Create key vault

Basics **Access policy** Networking Tags Review + create

Enable Access to:

- Azure Virtual Machines for deployment ⓘ
- Azure Resource Manager for template deployment ⓘ
- Azure Disk Encryption for volume encryption ⓘ

11. Select **Review + create**.
12. After the key vault has passed validation, select **Create**. This will return you to the **Select key from Azure Key Vault** screen.
13. Leave the **Key** field blank and choose **Select**.
14. At the top of the encryption screen, click **Save**. A popup will warn you that the VM will reboot. Click **Yes**.

## Clean up resources

When no longer needed, you can delete the resource group, virtual machine, and all related resources. To do so, select the resource group for the virtual machine, select **Delete**, then confirm the name of the resource group to delete.

## Next steps

In this quickstart, you created a Key Vault that was enabled for encryption keys, created a virtual machine, and enabled the virtual machine for encryption.

[Azure Disk Encryption overview](#)

# Azure Disk Encryption scenarios on Linux VMs

9/21/2022 • 19 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure Disk Encryption for Linux virtual machines (VMs) uses the DM-Crypt feature of Linux to provide full disk encryption of the OS disk and data disks. Additionally, it provides encryption of the temporary disk when using the EncryptFormatAll feature.

Azure Disk Encryption is [integrated with Azure Key Vault](#) to help you control and manage the disk encryption keys and secrets. For an overview of the service, see [Azure Disk Encryption for Linux VMs](#).

You can only apply disk encryption to virtual machines of [supported VM sizes and operating systems](#). You must also meet the following prerequisites:

- [Additional requirements for VMs](#)
- [Networking requirements](#)
- [Encryption key storage requirements](#)

In all cases, you should [take a snapshot](#) and/or create a backup before disks are encrypted. Backups ensure that a recovery option is possible if an unexpected failure occurs during encryption. VMs with managed disks require a backup before encryption occurs. Once a backup is made, you can use the [Set-AzVMDiskEncryptionExtension cmdlet](#) to encrypt managed disks by specifying the -skipVmBackup parameter. For more information about how to back up and restore encrypted VMs, see the [Azure Backup](#) article.

## WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM. See [Azure Disk Encryption with Azure AD \(previous release\)](#) for details.
- When encrypting Linux OS volumes, the VM should be considered unavailable. We strongly recommend to avoid SSH logins while the encryption is in progress to avoid issues blocking any open files that will need to be accessed during the encryption process. To check progress, use the the [Get-AzVMDiskEncryptionStatus](#) PowerShell cmdlet or the [vm encryption show](#) CLI command. This process can be expected to take a few hours for a 30GB OS volume, plus additional time for encrypting data volumes. Data volume encryption time will be proportional to the size and quantity of the data volumes unless the encrypt format all option is used.
- Disabling encryption on Linux VMs is only supported for data volumes. It is not supported on data or OS volumes if the OS volume has been encrypted.

## Install tools and connect to Azure

Azure Disk Encryption can be enabled and managed through the [Azure CLI](#) and [Azure PowerShell](#). To do so, you must install the tools locally and connect to your Azure subscription.

### Azure CLI

The [Azure CLI 2.0](#) is a command-line tool for managing Azure resources. The CLI is designed to flexibly query data, support long-running operations as non-blocking processes, and make scripting easy. You can install it locally by following the steps in [Install the Azure CLI](#).

To [Sign in to your Azure account with the Azure CLI](#), use the `az login` command.

```
az login
```

If you would like to select a tenant to sign in under, use:

```
az login --tenant <tenant>
```

If you have multiple subscriptions and want to specify a specific one, get your subscription list with [az account list](#) and specify with [az account set](#).

```
az account list  
az account set --subscription "<subscription name or ID>"
```

For more information, see [Get started with Azure CLI 2.0](#).

## Azure PowerShell

The [Azure PowerShell az module](#) provides a set of cmdlets that uses the [Azure Resource Manager](#) model for managing your Azure resources. You can use it in your browser with [Azure Cloud Shell](#), or you can install it on your local machine using the instructions in [Install the Azure PowerShell module](#).

If you already have it installed locally, make sure you use the latest version of Azure PowerShell SDK version to configure Azure Disk Encryption. Download the latest version of [Azure PowerShell release](#).

To [Sign in to your Azure account with Azure PowerShell](#), use the `Connect-AzAccount` cmdlet.

```
Connect-AzAccount
```

If you have multiple subscriptions and want to specify one, use the `Get-AzSubscription` cmdlet to list them, followed by the `Set-AzContext` cmdlet:

```
Set-AzContext -Subscription <SubscriptionId>
```

Running the `Get-AzContext` cmdlet will verify that the correct subscription has been selected.

To confirm the Azure Disk Encryption cmdlets are installed, use the `Get-command` cmdlet:

```
Get-command *diskencryption*
```

For more information, see [Getting started with Azure PowerShell](#).

## Enable encryption on an existing or running Linux VM

In this scenario, you can enable encryption by using the Resource Manager template, PowerShell cmdlets, or CLI commands. If you need schema information for the virtual machine extension, see the [Azure Disk Encryption for Linux extension](#) article.

## IMPORTANT

It is mandatory to snapshot and/or backup a managed disk based VM instance outside of, and prior to enabling Azure Disk Encryption. A snapshot of the managed disk can be taken from the portal, or through [Azure Backup](#). Backups ensure that a recovery option is possible in the case of any unexpected failure during encryption. Once a backup is made, the Set-AzVMDiskEncryptionExtension cmdlet can be used to encrypt managed disks by specifying the -skipVmBackup parameter. The Set-AzVMDiskEncryptionExtension command will fail against managed disk based VMs until a backup has been made and this parameter has been specified.

Encrypting or disabling encryption may cause the VM to reboot.

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

### Enable encryption on an existing or running Linux VM using Azure CLI

You can enable disk encryption on your encrypted VHD by installing and using the [Azure CLI](#) command-line tool. You can use it in your browser with [Azure Cloud Shell](#), or you can install it on your local machine and use it in any PowerShell session. To enable encryption on existing or running Linux VMs in Azure, use the following CLI commands:

Use the `az vm encryption enable` command to enable encryption on a running virtual machine in Azure.

- **Encrypt a running VM:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-encryption-keyvault "MySecureVault" --volume-type [All|OS|Data]
```

- **Encrypt a running VM using KEK:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type [All|OS|Data]
```

## NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: [https://\[keyvault-name\].vault.azure.net/keys/\[kekname\]/\[kek-unique-id\]](https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id])

- **Verify the disks are encrypted:** To check on the encryption status of a VM, use the `az vm encryption show` command.

```
az vm encryption show --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup"
```

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

### Enable encryption on an existing or running Linux VM using PowerShell

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet to enable encryption on a running virtual machine in Azure. Take a [snapshot](#) and/or back up the VM with [Azure Backup](#) before disks are encrypted. The -skipVmBackup parameter is already specified in the PowerShell scripts to encrypt a running Linux VM.

- **Encrypt a running VM:** The script below initializes your variables and runs the Set-

AzVMDiskEncryptionExtension cmdlet. The resource group, VM, and key vault, were created as prerequisites. Replace MyVirtualMachineResourceGroup, MySecureVM, and MySecureVault with your values. Modify the -VolumeType parameter to specify which disks you're encrypting.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyId $keyVaultResourceId -
VolumeType '[All|OS|Data]' -SequenceVersion $sequenceVersion -skipVmBackup;
```

- **Encrypt a running VM using KEK:** You may need to add the -VolumeType parameter if you're encrypting data disks and not the OS disk.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyId $keyVaultResourceId -
KeyEncryptionKeyUrl $keyEncryptionKeyUrl -KeyEncryptionKeyId $keyVaultResourceId -VolumeType
'[All|OS|Data]' -SequenceVersion $sequenceVersion -skipVmBackup;
```

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: [https://\[keyvault-name\].vault.azure.net/keys/\[kekname\]/\[kek-unique-id\]](https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id])

- **Verify the disks are encrypted:** To check on the encryption status of a VM, use the [Get-AzVmDiskEncryptionStatus](#) cmdlet.

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

#### Enable encryption on an existing or running Linux VM with a template

You can enable disk encryption on an existing or running Linux VM in Azure by using the [Resource Manager template](#).

1. Click Deploy to Azure on the Azure quickstart template.

- Select the subscription, resource group, resource group location, parameters, legal terms, and agreement.
- Click **Create** to enable encryption on the existing or running VM.

The following table lists Resource Manager template parameters for existing or running VMs:

PARAMETER	DESCRIPTION
vmName	Name of the VM to run the encryption operation.
keyVaultName	<p>Name of the key vault that the encryption key should be uploaded to. You can get it by using the cmdlet</p> <pre>(Get-AzKeyVault -ResourceGroupName &lt;MyKeyVaultResourceGroupName&gt;).Vaultname</pre> <p>or the Azure CLI command</p> <pre>az keyvault list --resource-group "MyKeyVaultResourceGroupName"</pre>
keyVaultResourceGroup	Name of the resource group that contains the key vault.
keyEncryptionKeyURL	<p>URL of the key encryption key that's used to encrypt the encryption key. This parameter is optional if you select <b>nokek</b> in the UseExistingKek drop-down list. If you select <b>kek</b> in the UseExistingKek drop-down list, you must enter the <i>keyEncryptionKeyURL</i> value.</p>
volumeType	Type of volume that the encryption operation is performed on. Valid values are <i>OS</i> , <i>Data</i> , and <i>All</i> .
forceUpdateTag	Pass in a unique value like a GUID every time the operation needs to be force run.
location	Location for all resources.

For more information about configuring the Linux VM disk encryption template, see [Azure Disk Encryption for Linux](#).

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

## Use EncryptFormatAll feature for data disks on Linux VMs

The **EncryptFormatAll** parameter reduces the time for Linux data disks to be encrypted. Partitions meeting certain criteria will be formatted, along with their current file systems, then remounted back to where they were before command execution. If you wish to exclude a data disk that meets the criteria, you can unmount it before running the command.

After running this command, any drives that were mounted previously will be reformatted, and the encryption layer will be started on top of the now empty drive. When this option is selected, the temporary disk attached to the VM will also be encrypted. If the temporary disk is reset, it will be reformatted and re-encrypted for the VM by the Azure Disk Encryption solution at the next opportunity. Once the resource disk gets encrypted, the [Microsoft Azure Linux Agent](#) will not be able to manage the resource disk and enable the swap file, but you may manually configure the swap file.

### WARNING

EncryptFormatAll shouldn't be used when there is needed data on a VM's data volumes. You may exclude disks from encryption by unmounting them. You should first try out the EncryptFormatAll first on a test VM, understand the feature parameter and its implication before trying it on the production VM. The EncryptFormatAll option formats the data disk and all the data on it will be lost. Before proceeding, verify that disks you wish to exclude are properly unmounted.

If you're setting this parameter while updating encryption settings, it might lead to a reboot before the actual encryption. In this case, you will also want to remove the disk you don't want formatted from the fstab file. Similarly, you should add the partition you want encrypt-formatted to the fstab file before initiating the encryption operation.

### EncryptFormatAll criteria

The parameter goes through all partitions and encrypts them as long as they meet **all** of the criteria below:

- Is not a root/OS/boot partition
- Is not already encrypted
- Is not a BEK volume
- Is not a RAID volume
- Is not an LVM volume
- Is mounted

Encrypt the disks that compose the RAID or LVM volume rather than the RAID or LVM volume.

### Use the EncryptFormatAll parameter with Azure CLI

Use the [az vm encryption enable](#) command to enable encryption on a running virtual machine in Azure.

- **Encrypt a running VM using EncryptFormatAll:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-encryption-keyvault "MySecureVault" --volume-type "data" --encrypt-format-all
```

### Use the EncryptFormatAll parameter with a PowerShell cmdlet

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet with the EncryptFormatAll parameter.

**Encrypt a running VM using EncryptFormatAll:** As an example, the script below initializes your variables and runs the Set-AzVMDiskEncryptionExtension cmdlet with the EncryptFormatAll parameter. The resource group, VM, and key vault were created as prerequisites. Replace MyVirtualMachineResourceGroup, MySecureVM, and MySecureVault with your values.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$KeyVaultName = 'MySecureVault';
$KeyVault = Get-AzKeyVault -VaultName $KeyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $KeyVault.VaultUri;
$keyVaultResourceId = $KeyVault.ResourceId;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -DiskEncryptionKeyVaultUrl
$diskEncryptionKeyVaultUrl -DiskEncryptionKeyId $keyVaultResourceId -VolumeType "data" -
EncryptFormatAll
```

### Use the EncryptFormatAll parameter with Logical Volume Manager (LVM)

We recommend an LVM-on-crypt setup. For all the following examples, replace the device-path and mountpoints with whatever suits your use-case. This setup can be done as follows:

1. Add the data disks that will compose the VM.
2. Format, mount, and add these disks to the fstab file.
3. Choose a partition standard, create a partition that spans the entire drive, and then format the partition.  
We use symlinks generated by Azure here. Using symlinks avoids problems related to device names changing. For more information, see the [Troubleshoot Device Names problems](#) article.

```
parted /dev/disk/azure/scsi1/lun0 mklabel gpt  
parted -a opt /dev/disk/azure/scsi1/lun0 mkpart primary ext4 0% 100%  
  
mkfs -t ext4 /dev/disk/azure/scsi1/lun0-part1
```

4. Mount the disks:

```
mount /dev/disk/azure/scsi1/lun0-part1 /mnt/mountpoint
```

Add to fstab file:

```
echo "/dev/disk/azure/scsi1/lun0-part1 /mnt/mountpoint ext4 defaults,nofail 0 2" >> /etc/fstab
```

5. Run the Azure PowerShell [Set-AzVMDiskEncryptionExtension](#) cmdlet with -EncryptFormatAll to encrypt these disks.

```
$KeyVault = Get-AzKeyVault -VaultName "MySecureVault" -ResourceGroupName "MySecureGroup"  
  
Set-AzVMDiskEncryptionExtension -ResourceGroupName "MySecureGroup" -VMName "MySecureVM" -  
DiskEncryptionKeyVaultUrl $KeyVault.VaultUri -DiskEncryptionKeyId $KeyVault.ResourceId -  
EncryptFormatAll -SkipVmBackup -VolumeType Data
```

If you wish to use a key encryption key (KEK), pass the URI of your KEK and the ResourceID of your key vault to the -KeyEncryptionKeyUrl and -KeyEncryptionKeyId parameters, respectively:

```
$KeyVault = Get-AzKeyVault -VaultName "MySecureVault" -ResourceGroupName "MySecureGroup"  
$KEKKeyVault = Get-AzKeyVault -VaultName "MyKEKVault" -ResourceGroupName "MySecureGroup"  
$KEK = Get-AzKeyVaultKey -VaultName "myKEKVault" -KeyName "myKEKName"  
  
Set-AzVMDiskEncryptionExtension -ResourceGroupName "MySecureGroup" -VMName "MySecureVM" -  
DiskEncryptionKeyVaultUrl $KeyVault.VaultUri -DiskEncryptionKeyId $KeyVault.ResourceId -  
EncryptFormatAll -SkipVmBackup -VolumeType Data -KeyEncryptionKeyUrl $$KEK.id -  
KeyEncryptionKeyId $KEKKeyVault.ResourceId
```

6. Set up LVM on top of these new disks. Note the encrypted drives are unlocked after the VM has finished booting. So, the LVM mounting will also have to be subsequently delayed.

## New VMs created from customer-encrypted VHD and encryption keys

In this scenario, you can enable encrypting by using PowerShell cmdlets or CLI commands.

Use the instructions in the Azure Disk encryption same scripts for preparing pre-encrypted images that can be used in Azure. After the image is created, you can use the steps in the next section to create an encrypted Azure VM.

- [Prepare a pre-encrypted Linux VHD](#)

## IMPORTANT

It is mandatory to snapshot and/or backup a managed disk based VM instance outside of, and prior to enabling Azure Disk Encryption. A snapshot of the managed disk can be taken from the portal, or [Azure Backup](#) can be used. Backups ensure that a recovery option is possible in the case of any unexpected failure during encryption. Once a backup is made, the Set-AzVMDiskEncryptionExtension cmdlet can be used to encrypt managed disks by specifying the -skipVmBackup parameter. The Set-AzVMDiskEncryptionExtension command will fail against managed disk based VMs until a backup has been made and this parameter has been specified.

Encrypting or disabling encryption may cause the VM to reboot.

## Use Azure PowerShell to encrypt VMs with pre-encrypted VHDs

You can enable disk encryption on your encrypted VHD by using the PowerShell cmdlet [Set-AzVMOSDisk](#). The example below gives you some common parameters.

```
$VirtualMachine = New-AzVMConfig -VMName "MySecureVM" -VMSize "Standard_A1"
$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name "SecureOSDisk" -VhdUri "os.vhd" Caching ReadWrite
-Linux -CreateOption "Attach" -DiskEncryptionKeyUrl
"https://mytestvault.vault.azure.net/secrets/Test1/514ceb769c984379a7e0230bddaaaaaaaa"
-DiskEncryptionKeyVaultId "/subscriptions/00000000-0000-0000-0000-
000000000000/resourceGroups/myresourcegroup/providers/Microsoft.KeyVault/vaults/mytestvault"
New-AzVM -VM $VirtualMachine -ResourceGroupName "MyVirtualMachineResourceGroup"
```

## Enable encryption on a newly added data disk

You can add a new data disk using [az vm disk attach](#), or [through the Azure portal](#). Before you can encrypt, you need to mount the newly attached data disk first. You must request encryption of the data drive since the drive will be unusable while encryption is in progress.

### Enable encryption on a newly added disk with Azure CLI

If the VM was previously encrypted with "All" then the --volume-type parameter should remain "All". All includes both OS and data disks. If the VM was previously encrypted with a volume type of "OS", then the --volume-type parameter should be changed to "All" so that both the OS and the new data disk will be included. If the VM was encrypted with only the volume type of "Data", then it can remain "Data" as demonstrated below. Adding and attaching a new data disk to a VM is not sufficient preparation for encryption. The newly attached disk must also be formatted and properly mounted within the VM prior to enabling encryption. On Linux the disk must be mounted in /etc/fstab with a [persistent block device name](#).

In contrast to PowerShell syntax, the CLI does not require the user to provide a unique sequence version when enabling encryption. The CLI automatically generates and uses its own unique sequence version value.

- Encrypt data volumes of a running VM:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-
encryption-keyvault "MySecureVault" --volume-type "Data"
```

- Encrypt data volumes of a running VM using KEK:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-
encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault
"MySecureVaultContainingTheKEK" --volume-type "Data"
```

### Enable encryption on a newly added disk with Azure PowerShell

When using PowerShell to encrypt a new disk for Linux, a new sequence version needs to be specified. The

sequence version has to be unique. The script below generates a GUID for the sequence version. Take a [snapshot](#) and/or back up the VM with [Azure Backup](#) before disks are encrypted. The `-skipVmBackup` parameter is already specified in the PowerShell scripts to encrypt a newly added data disk.

- **Encrypt data volumes of a running VM:** The script below initializes your variables and runs the `Set-AzVMDiskEncryptionExtension` cmdlet. The resource group, VM, and key vault should have already been created as prerequisites. Replace `MyVirtualMachineResourceGroup`, `MySecureVM`, and `MySecureVault` with your values. Acceptable values for the `-VolumeType` parameter are All, OS, and Data. If the VM was previously encrypted with a volume type of "OS" or "All", then the `-VolumeType` parameter should be changed to "All" so that both the OS and the new data disk will be included.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyVaultId $keyVaultResourceId -
VolumeType 'data' -SequenceVersion $sequenceVersion -skipVmBackup;
```

- **Encrypt data volumes of a running VM using KEK:** Acceptable values for the `-VolumeType` parameter are All, OS, and Data. If the VM was previously encrypted with a volume type of "OS" or "All", then the `-VolumeType` parameter should be changed to All so that both the OS and the new data disk will be included.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyVaultId $keyVaultResourceId -
KeyEncryptionKeyUrl $keyEncryptionKeyUrl -KeyEncryptionKeyVaultId $keyVaultResourceId -VolumeType
'data' -SequenceVersion $sequenceVersion -skipVmBackup;
```

#### NOTE

The syntax for the value of `disk-encryption-keyvault` parameter is the full identifier string:  
`/subscriptions/[subscription-id-guid]/resourceGroups/[KVresource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]`

The syntax for the value of the `key-encryption-key` parameter is the full URI to the KEK as in: `https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]`

## Disable encryption and remove the encryption extension

You can disable the Azure disk encryption extension, and you can remove the Azure disk encryption extension.

These are two distinct operations.

To remove ADE, it is recommended that you first disable encryption and then remove the extension. If you remove the encryption extension without disabling it, the disks will still be encrypted. If you disable encryption after removing the extension, the extension will be reinstalled (to perform the decrypt operation) and will need to be removed a second time.

#### WARNING

You can **not** disable encryption if the OS disk is encrypted. (OS disks are encrypted when the original encryption operation specifies `volumeType=ALL` or `volumeType=OS`.)

Disabling encryption works only when data disks are encrypted but the OS disk is not.

## Disable encryption

You can disable encryption using Azure PowerShell, the Azure CLI, or with a Resource Manager template.

Disabling encryption does **not** remove the extension (see [Remove the encryption extension](#)).

- **Disable disk encryption with Azure PowerShell:** To disable the encryption, use the [Disable-AzVMDiskEncryption](#) cmdlet.

```
Disable-AzVMDiskEncryption -ResourceGroupName "MyVirtualMachineResourceGroup" -VMName "MySecureVM" -VolumeType "data"
```

- **Disable encryption with the Azure CLI:** To disable encryption, use the [az vm encryption disable](#) command.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --volume-type "data"
```

- **Disable encryption with a Resource Manager template:**

1. Click **Deploy to Azure** from the [Disable disk encryption on running Linux VM](#) template.
2. Select the subscription, resource group, location, VM, volume type, legal terms, and agreement.
3. Click **Purchase** to disable disk encryption on a running Linux VM.

## Remove the encryption extension

If you want to decrypt your disks and remove the encryption extension, you must disable encryption **before** removing the extension; see [disable encryption](#).

You can remove the encryption extension using Azure PowerShell or the Azure CLI.

- **Disable disk encryption with Azure PowerShell:** To remove the encryption, use the [Remove-AzVMDiskEncryptionExtension](#) cmdlet.

```
Remove-AzVMDiskEncryptionExtension -ResourceGroupName "MyVirtualMachineResourceGroup" -VMName "MySecureVM"
```

- **Disable encryption with the Azure CLI:** To remove encryption, use the [az vm extension delete](#) command.

```
az vm extension delete -g "MyVirtualMachineResourceGroup" --vm-name "MySecureVM" -n "AzureDiskEncryptionForLinux"
```

## Unsupported scenarios

Azure Disk Encryption does not work for the following Linux scenarios, features, and technology:

- Encrypting basic tier VM or VMs created through the classic VM creation method.
- Disabling encryption on an OS drive or data drive of a Linux VM when the OS drive is encrypted.
- Encrypting the OS drive for Linux virtual machine scale sets.
- Encrypting custom images on Linux VMs.
- Integration with an on-premises key management system.
- Azure Files (shared file system).
- Network File System (NFS).
- Dynamic volumes.
- Ephemeral OS disks.
- Encryption of shared/distributed file systems like (but not limited to): DFS, GFS, DRDB, and CephFS.
- Moving an encrypted VM to another subscription or region.
- Creating an image or snapshot of an encrypted VM and using it to deploy additional VMs.
- Kernel Crash Dump (kdump).
- Oracle ACFS (ASM Cluster File System).
- NVMe disks such as those on [High performance computing VM sizes](#) or [Storage optimized VM sizes](#).
- A VM with "nested mount points"; that is, multiple mount points in a single path (such as "/1stmountpoint/data/2stmountpoint").
- A VM with a data drive mounted on top of an OS folder.
- A VM on which a root (OS disk) logical volume has been extended using a data disk.
- M-series VMs with Write Accelerator disks.
- Applying ADE to a VM that has disks encrypted with [server-side encryption with customer-managed keys](#) (SSE + CMK). Applying SSE + CMK to a data disk on a VM encrypted with ADE is an unsupported scenario as well.
- Migrating a VM that is encrypted with ADE, or has ever been encrypted with ADE, to [server-side encryption with customer-managed keys](#).
- Encrypting VMs in failover clusters.
- Encryption of [Azure ultra disks](#).

## Next steps

- [Azure Disk Encryption overview](#)
- [Azure Disk Encryption sample scripts](#)
- [Azure Disk Encryption troubleshooting](#)

# Creating and configuring a key vault for Azure Disk Encryption

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure Disk Encryption uses Azure Key Vault to control and manage disk encryption keys and secrets. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#).

## WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM. See [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#) for details.

Creating and configuring a key vault for use with Azure Disk Encryption involves three steps:

1. Creating a resource group, if needed.
2. Creating a key vault.
3. Setting key vault advanced access policies.

These steps are illustrated in the following quickstarts:

- [Create and encrypt a Linux VM with Azure CLI](#)
- [Create and encrypt a Linux VM with Azure PowerShell](#)

You may also, if you wish, generate or import a key encryption key (KEK).

## NOTE

The steps in this article are automated in the [Azure Disk Encryption prerequisites CLI script](#) and [Azure Disk Encryption prerequisites PowerShell script](#).

## Install tools and connect to Azure

The steps in this article can be completed with the [Azure CLI](#), the [Azure PowerShell Az module](#), or the [Azure portal](#).

While the portal is accessible through your browser, Azure CLI and Azure PowerShell require local installation; see [Azure Disk Encryption for Linux: Install tools](#) for details.

### Connect to your Azure account

Before using the Azure CLI or Azure PowerShell, you must first connect to your Azure subscription. You do so by [Signing in with Azure CLI](#), [Signing in with Azure PowerShell](#), or supplying your credentials to the Azure portal when prompted.

```
az login
```

```
Connect-AzAccount
```

## Create a resource group

If you already have a resource group, you can skip to [Create a key vault](#).

A resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group using the [az group create](#) Azure CLI command, the [New-AzResourceGroup](#) Azure PowerShell command, or from the [Azure portal](#).

- [Azure portal](#)

### Azure CLI

```
az group create --name "myResourceGroup" --location eastus
```

### Azure PowerShell

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

## Create a key vault

If you already have a key vault, you can skip to [Set key vault advanced access policies](#).

Create a key vault using the [az keyvault create](#) Azure CLI command, the [New-AzKeyvault](#) Azure PowerShell command, the [Azure portal](#), or a [Resource Manager template](#).

#### WARNING

To ensure that encryption secrets don't cross regional boundaries, you must create and use a key vault that's in the **same region and tenant** as the VMs to be encrypted.

Each Key Vault must have a unique name. Replace <your-unique-keyvault-name> with the name of your key vault in the following examples.

### Azure CLI

When creating a key vault by using the Azure CLI, add the "--enabled-for-disk-encryption" flag.

```
az keyvault create --name "<your-unique-keyvault-name>" --resource-group "myResourceGroup" --location "eastus" --enabled-for-disk-encryption
```

### Azure PowerShell

When creating a key vault using Azure PowerShell, add the "-EnabledForDiskEncryption" flag.

```
New-AzKeyvault -name "<your-unique-keyvault-name>" -ResourceGroupName "myResourceGroup" -Location "eastus" -EnabledForDiskEncryption
```

### Resource Manager template

You can also create a key vault by using the [Resource Manager template](#).

1. On the Azure Quickstart Template, click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, Key Vault name, Object ID, legal terms, and agreement, and then click **Purchase**.

## Set key vault advanced access policies

### IMPORTANT

Newly-created key vaults have soft-delete on by default. If you are using a pre-existing key vault, you **must** enable soft-delete. See [Azure Key Vault soft-delete overview](#).

The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the VM for booting and decrypting the volumes.

If you didn't enable your key vault for disk encryption, deployment, or template deployment at the time of creation (as demonstrated in the previous step), you must update its advanced access policies.

### Azure CLI

Use [az keyvault update](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** Enabled-for-disk-encryption is required.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-disk-encryption "true"
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-deployment "true"
```

- **Enable Key Vault for template deployment, if needed:** Allow Resource Manager to retrieve secrets from the vault.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-template-deployment "true"
```

### Azure PowerShell

Use the key vault PowerShell cmdlet [Set-AzKeyVaultAccessPolicy](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** EnabledForDiskEncryption is required for Azure Disk encryption.

```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForDiskEncryption
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

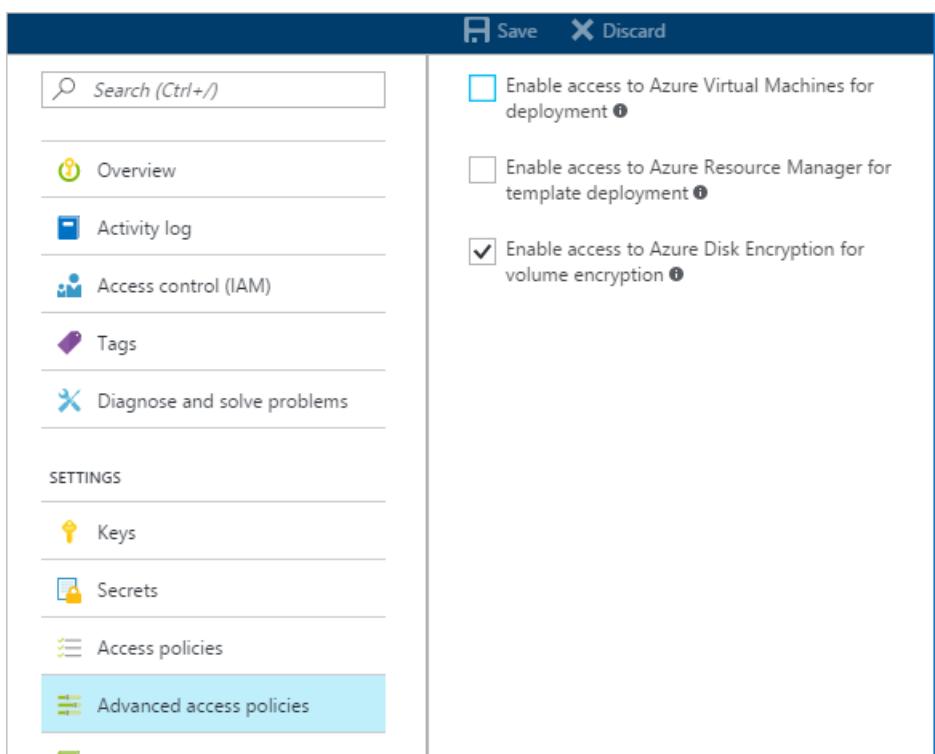
```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForDeployment
```

- **Enable Key Vault for template deployment, if needed:** Enables Azure Resource Manager to get secrets from this key vault when this key vault is referenced in a template deployment.

```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForTemplateDeployment
```

## Azure portal

1. Select your key vault and go to **Access Policies**.
2. Under "Enable Access to", select the box labeled **Azure Disk Encryption for volume encryption**.
3. Select **Azure Virtual Machines for deployment** and/or **Azure Resource Manager for template deployment**, if needed.
4. Click **Save**.



## Azure Disk Encryption and auto-rotation

Although Azure Key Vault now has [key auto-rotation](#), it isn't currently compatible with Azure Disk Encryption. Specifically, Azure Disk Encryption will continue to use the original encryption key, even after it has been auto-rotated.

Rotating an encryption key won't break Azure Disk Encryption, but disabling the "old" encryption key (in other words, the key Azure Disk Encryption is still using) will.

## Set up a key encryption key (KEK)

### IMPORTANT

The account running to enable disk encryption over the key vault must have "reader" permissions.

If you want to use a key encryption key (KEK) for an additional layer of security for encryption keys, add a KEK to your key vault. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault.

You can generate a new KEK by using the Azure CLI `az keyvault key create` command, the Azure PowerShell `Add-AzKeyVaultKey` cmdlet, or the [Azure portal](#). You must generate an RSA key type; Azure Disk Encryption doesn't currently support using Elliptic Curve keys.

You can instead import a KEK from your on-premises key management HSM. For more information, see [Key Vault Documentation](#).

Your key vault KEK URLs must be versioned. Azure enforces this restriction of versioning. For valid secret and KEK URLs, see the following examples:

- Example of a valid secret URL:

<https://contosovault.vault.azure.net/secrets/EncryptionSecretWithKek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

- Example of a valid KEK URL:

<https://contosovault.vault.azure.net/keys/diskencryptionkek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

## Azure CLI

Use the Azure CLI `az keyvault key create` command to generate a new KEK and store it in your key vault.

```
az keyvault key create --name "myKEK" --vault-name "<your-unique-keyvault-name>" --kty RSA --size 4096
```

You may instead import a private key by using the Azure CLI `az keyvault key import` command:

In either case, you supply the name of your KEK to the Azure CLI `az vm encryption enable` `--key-encryption-key` parameter.

```
az vm encryption enable -g "MyResourceGroup" --name "myVM" --disk-encryption-keyvault "<your-unique-keyvault-name>" --key-encryption-key "myKEK"
```

## Azure PowerShell

Use the Azure PowerShell `Add-AzKeyVaultKey` cmdlet to generate a new KEK and store it in your key vault.

```
Add-AzKeyVaultKey -Name "myKEK" -VaultName "<your-unique-keyvault-name>" -Destination "HSM" -Size 4096
```

You may instead import a private key using the Azure PowerShell `az keyvault key import` command.

In either case, you will supply the ID of your KEK key Vault and the URL of your KEK to the Azure PowerShell `Set-AzVMDiskEncryptionExtension` `-KeyEncryptionKeyVaultId` and `-KeyEncryptionKeyUrl` parameters. This example assumes that you are using the same key vault for both the disk encryption key and the KEK.

```
$KeyVault = Get-AzKeyVault -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "myResourceGroup"
$KEK = Get-AzKeyVaultKey -VaultName "<your-unique-keyvault-name>" -Name "myKEK"

Set-AzVMDiskEncryptionExtension -ResourceGroupName MyResourceGroup -VMName "MyVM" -DiskEncryptionKeyVaultUrl
$KeyVault.VaultUri -DiskEncryptionKeyVaultId $KeyVault.ResourceId -KeyEncryptionKeyVaultId
$KeyVault.ResourceId -KeyEncryptionKeyUrl $KEK.Id -SkipVmBackup -VolumeType All
```

## Next steps

- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)

- Learn Azure Disk Encryption scenarios on Linux VMs
- Learn how to troubleshoot Azure Disk Encryption
- Read the [Azure Disk Encryption sample scripts](#)

# Azure Disk Encryption sample scripts for Linux VMs

9/21/2022 • 13 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article provides sample scripts for preparing pre-encrypted VHDs and other tasks.

## NOTE

All scripts refer to the latest, non-AAD version of ADE, except where noted.

## Sample PowerShell scripts for Azure Disk Encryption

- **List all encrypted VMs in your subscription**

You can find all ADE-encrypted VMs and the extension version, in all resource groups present in a subscription, using [this PowerShell script](#).

Alternatively, these cmdlets will show all ADE-encrypted VMs (but not the extension version):

```
$osVolEncrypted = {(Get-AzVMDiskEncryptionStatus -ResourceGroupName $_.ResourceGroupName -VMName
$_.Name).OsVolumeEncrypted}
$dataVolEncrypted= {(Get-AzVMDiskEncryptionStatus -ResourceGroupName $_.ResourceGroupName -VMName
$_.Name).DataVolumesEncrypted}
Get-AzVm | Format-Table @{Label="MachineName"; Expression={$.Name}}, @{Label="OsVolumeEncrypted";
Expression=$osVolEncrypted}, @{Label="DataVolumesEncrypted"; Expression=$dataVolEncrypted}
```

- **List all encrypted VMSS instances in your subscription**

You can find all ADE-encrypted VMSS instances and the extension version, in all resource groups present in a subscription, using [this PowerShell script](#).

- **List all disk encryption secrets used for encrypting VMs in a key vault**

```
Get-AzKeyVaultSecret -VaultName $KeyVaultName | where
{$_.Tags.ContainsKey('DiskEncryptionKeyFileName')} | format-table @{Label="MachineName"; Expression=
{$_.Tags['MachineName']}}, @{Label="VolumeLetter"; Expression={$_.Tags['VolumeLetter']}},
@{Label="EncryptionKeyURL"; Expression={$_.Id}}
```

## Using the Azure Disk Encryption prerequisites PowerShell script

If you're already familiar with the prerequisites for Azure Disk Encryption, you can use the [Azure Disk Encryption prerequisites PowerShell script](#). For an example of using this PowerShell script, see the [Encrypt a VM Quickstart](#). You can remove the comments from a section of the script, starting at line 211, to encrypt all disks for existing VMs in an existing resource group.

The following table shows which parameters can be used in the PowerShell script:

PARAMETER	DESCRIPTION	MANDATORY?
-----------	-------------	------------

PARAMETER	DESCRIPTION	MANDATORY?
\$resourceGroupName	Name of the resource group to which the KeyVault belongs to. A new resource group with this name will be created if one doesn't exist.	True
\$keyVaultName	Name of the KeyVault in which encryption keys are to be placed. A new vault with this name will be created if one doesn't exist.	True
\$location	Location of the KeyVault. Make sure the KeyVault and VMs to be encrypted are in the same location. Get a location list with <a href="#">Get-AzLocation</a> .	True
\$subscriptionId	Identifier of the Azure subscription to be used. You can get your Subscription ID with <a href="#">Get-AzSubscription</a> .	True
\$aadAppName	Name of the Azure AD application that will be used to write secrets to KeyVault. A new application with this name will be created if one doesn't exist. If this app already exists, pass aadClientSecret parameter to the script.	False
\$aadClientSecret	Client secret of the Azure AD application that was created earlier.	False
\$keyEncryptionKeyName	Name of optional key encryption key in KeyVault. A new key with this name will be created if one doesn't exist.	False

### Encrypt or decrypt VMs without an Azure AD app

- [Enable disk encryption on an existing or running Linux VM](#)
- [Disable encryption on a running Linux VM](#)
  - Disabling encryption is only allowed on Data volumes for Linux VMs.

### Encrypt or decrypt VMs with an Azure AD app (previous release)

- [Enable disk encryption on an existing or running Linux VM](#)
- [Disable encryption on a running Linux VM](#)
  - Disabling encryption is only allowed on Data volumes for Linux VMs.
- [Create a new encrypted managed disk from a pre-encrypted VHD/storage blob](#)
  - Creates a new encrypted managed disk provided a pre-encrypted VHD and its corresponding encryption settings

## Encrypting an OS drive on a running Linux VM

### Prerequisites for OS disk encryption

- The VM must be using a distribution compatible with OS disk encryption as listed in the [Azure Disk Encryption supported operating systems](#)

- The VM must be created from the Marketplace image in Azure Resource Manager.
- Azure VM with at least 4 GB of RAM (recommended size is 7 GB).
- (For RHEL and CentOS) Disable SELinux. To disable SELinux, see "4.4.2. Disabling SELinux" in the [SELinux User's and Administrator's Guide](#) on the VM.
- After you disable SELinux, reboot the VM at least once.

## Steps

1. Create a VM by using one of the distributions specified previously.

For CentOS 7.2, OS disk encryption is supported via a special image. To use this image, specify "7.2n" as the SKU when you create the VM:

```
Set-AzVMSourceImage -VM $VirtualMachine -PublisherName "OpenLogic" -Offer "CentOS" -Skus "7.2n" - Version "latest"
```

2. Configure the VM according to your needs. If you're going to encrypt all the (OS + data) drives, the data drives need to be specified and mountable from /etc/fstab.

### NOTE

Use UUID=... to specify data drives in /etc/fstab instead of specifying the block device name (for example, /dev/sdb1). During encryption, the order of drives changes on the VM. If your VM relies on a specific order of block devices, it will fail to mount them after encryption.

3. Sign out of the SSH sessions.

4. To encrypt the OS, specify volumeType as All or OS when you enable encryption.

### NOTE

All user-space processes that are not running as `systemd` services should be killed with a `SIGKILL`. Reboot the VM. When you enable OS disk encryption on a running VM, plan on VM downtime.

5. Periodically monitor the progress of encryption by using the instructions in the [next section](#).

6. After `Get-AzVmDiskEncryptionStatus` shows "VMRestartPending", restart your VM either by signing in to it or by using the portal, PowerShell, or CLI.

```
C:\> Get-AzVmDiskEncryptionStatus -ResourceGroupName $ResourceGroupName -VMName $VMName -ExtensionName $ExtensionName

OsVolumeEncrypted      : VMRestartPending
DataVolumesEncrypted   : NotMounted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : OS disk successfully encrypted, reboot the VM
```

Before you reboot, we recommend that you save [boot diagnostics](#) of the VM.

## Monitoring OS encryption progress

You can monitor OS encryption progress in three ways:

- Use the `Get-AzVmDiskEncryptionStatus` cmdlet and inspect the ProgressMessage field:

```

OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : NotMounted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage        : OS disk encryption started

```

After the VM reaches "OS disk encryption started", it takes about 40 to 50 minutes on a Premium-storage backed VM.

Because of [issue #388](#) in WALinuxAgent, `osVolumeEncrypted` and `DataVolumesEncrypted` show up as `Unknown` in some distributions. With WALinuxAgent version 2.1.5 and later, this issue is fixed automatically. If you see `Unknown` in the output, you can verify disk-encryption status by using the Azure Resource Explorer.

Go to [Azure Resource Explorer](#), and then expand this hierarchy in the selection panel on left:

```

|-- subscriptions
  |-- [Your subscription]
    |-- resourceGroups
      |-- [Your resource group]
        |-- providers
          |-- Microsoft.Compute
            |-- virtualMachines
              |-- [Your virtual machine]
                |-- InstanceView

```

In the InstanceView, scroll down to see the encryption status of your drives.

```

{
  "code": "ProvisioningState/succeeded",
  "level": "Info",
  "displayStatus": "Provisioning succeeded",
  "time": "2016-09-22T02:19:41.4646766+00:00"
},
],
"extensions": [
  {
    "name": "AzureDiskEncryptionForLinux",
    "type": "Microsoft.Azure.Security.AzureDiskEncryptionForLinux",
    "typeHandlerVersion": "0.1.0.999190",
    "substatuses": [
      {
        "code": "ComponentStatus/Microsoft.Azure.Security.AzureDiskEncryptionForLinux",
        "level": "Info",
        "displayStatus": "Provisioning succeeded",
        "message": "{\"os\": \"NotEncrypted\", \"data\": \"EncryptionInProgress\"}"
      }
    ]
  }
]
}

```

- Look at [boot diagnostics](#). Messages from the ADE extension should be prefixed with `[AzureDiskEncryption]`.
- Sign in to the VM via SSH, and get the extension log from:

`/var/log/azure/Microsoft.Azure.Security.AzureDiskEncryptionForLinux`

We recommend that you don't sign-in to the VM while OS encryption is in progress. Copy the logs only when the other two methods have failed.

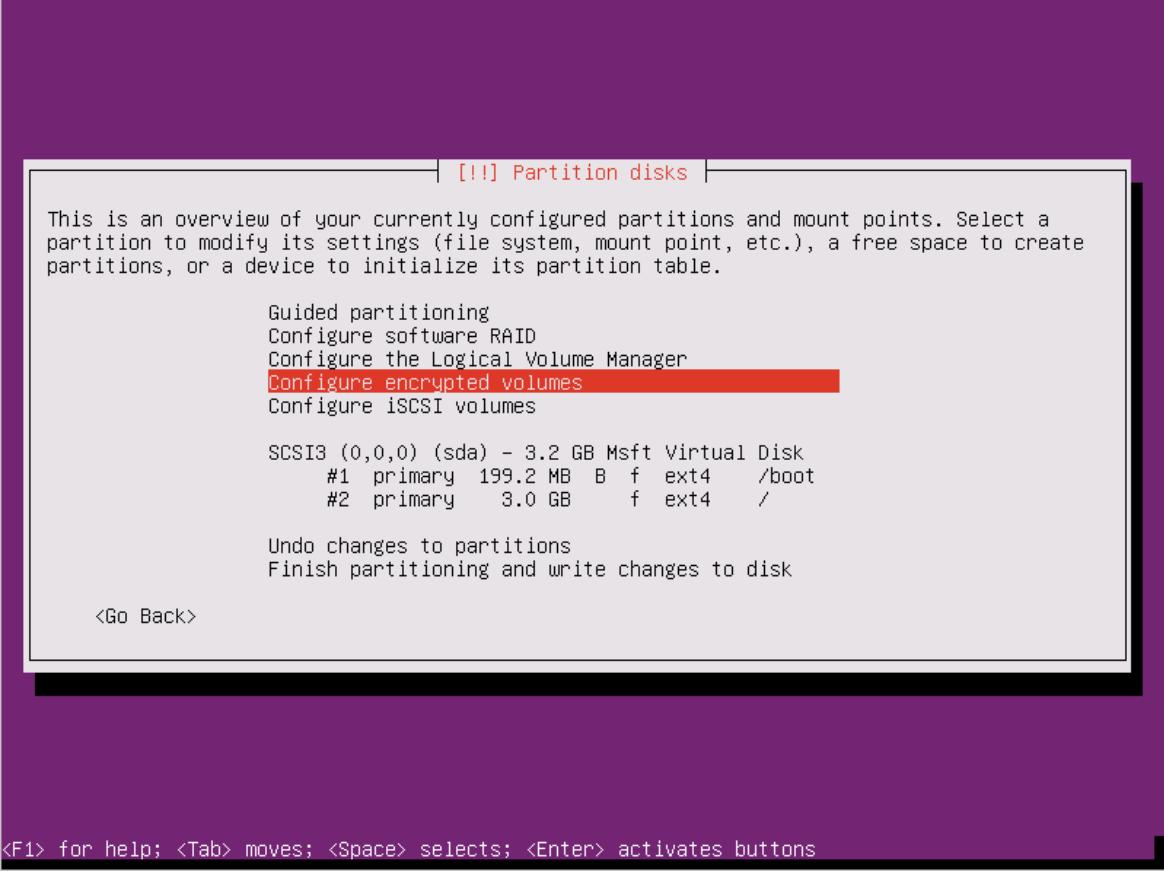
## Prepare a pre-encrypted Linux VHD

The preparation for pre-encrypted VHDs can vary depending on the distribution. Examples on preparing Ubuntu 16, openSUSE 13.2, and CentOS 7 are available.

### Ubuntu 16

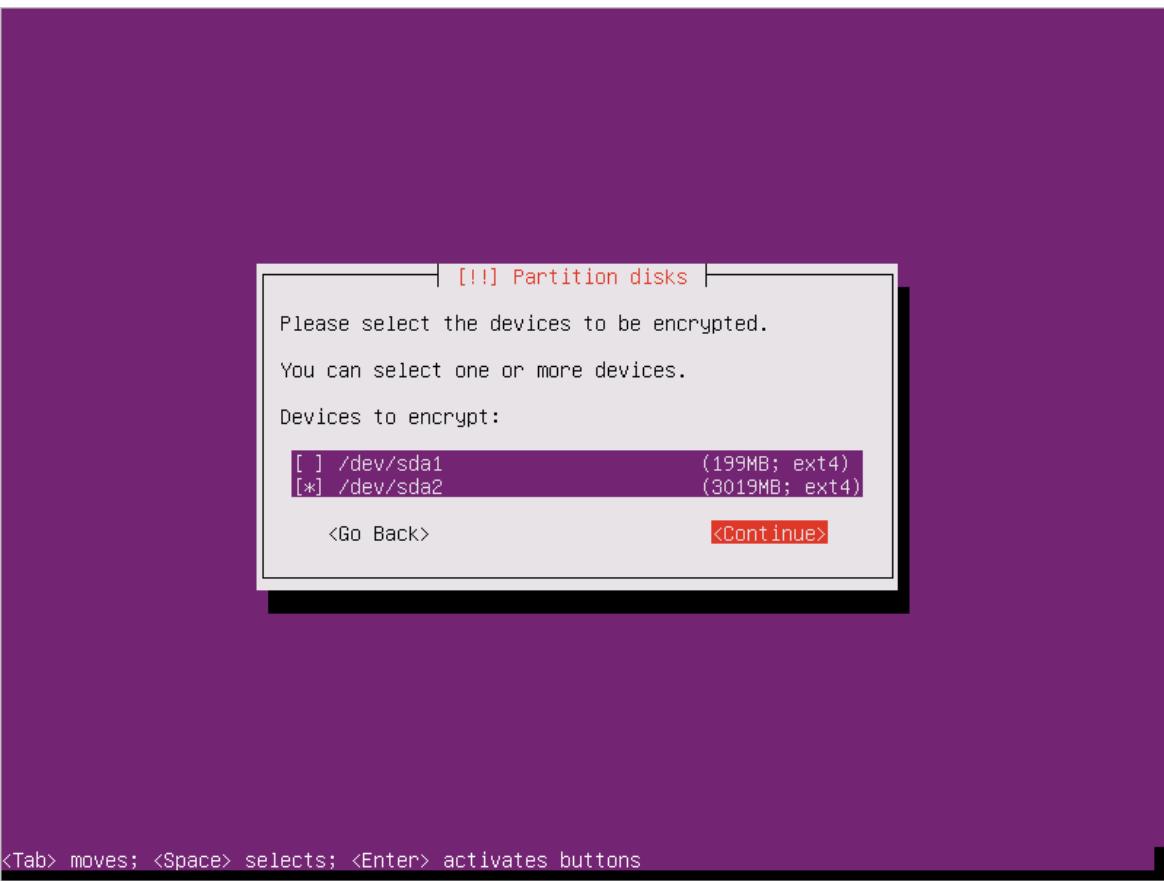
Configure encryption during the distribution installation by doing the following steps:

- Select **Configure encrypted volumes** when you partition the disks.



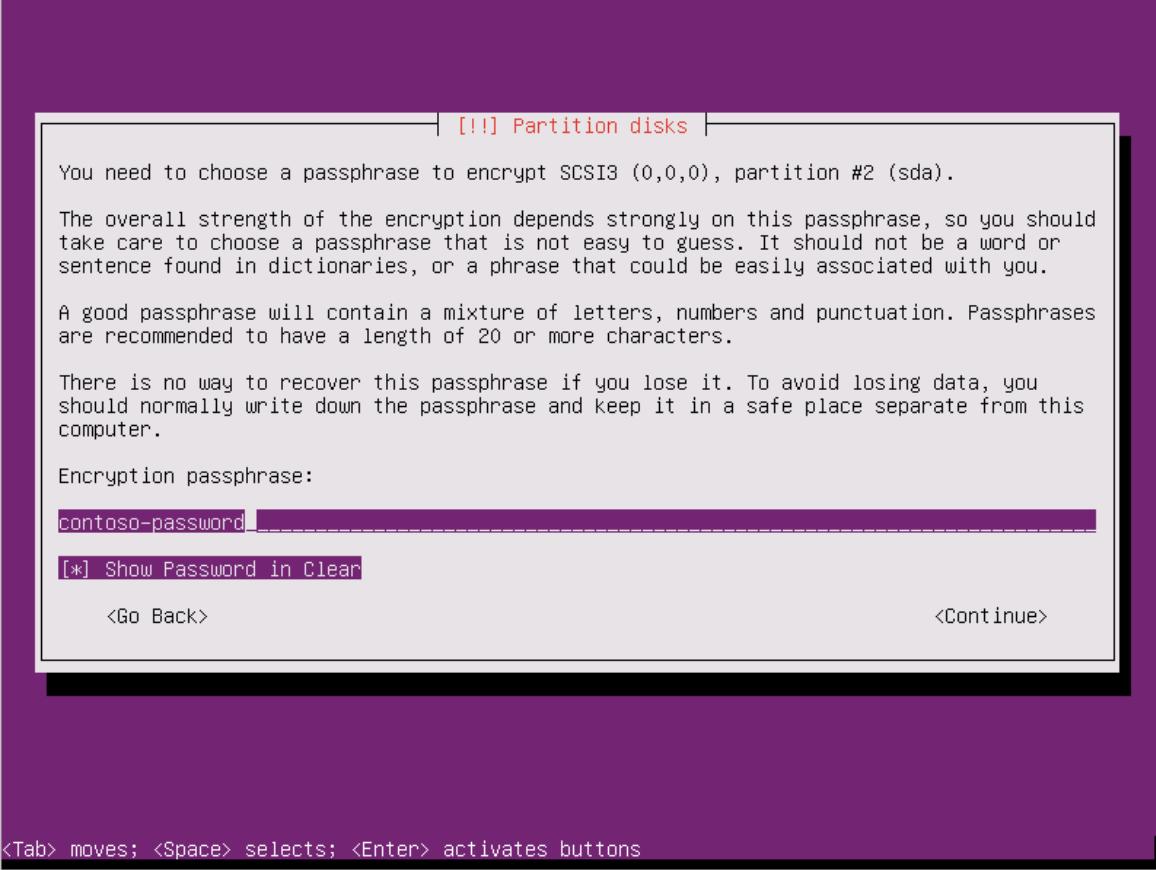
<F1> for help; <Tab> moves; <Space> selects; <Enter> activates buttons

2. Create a separate boot drive, which must not be encrypted. Encrypt your root drive.

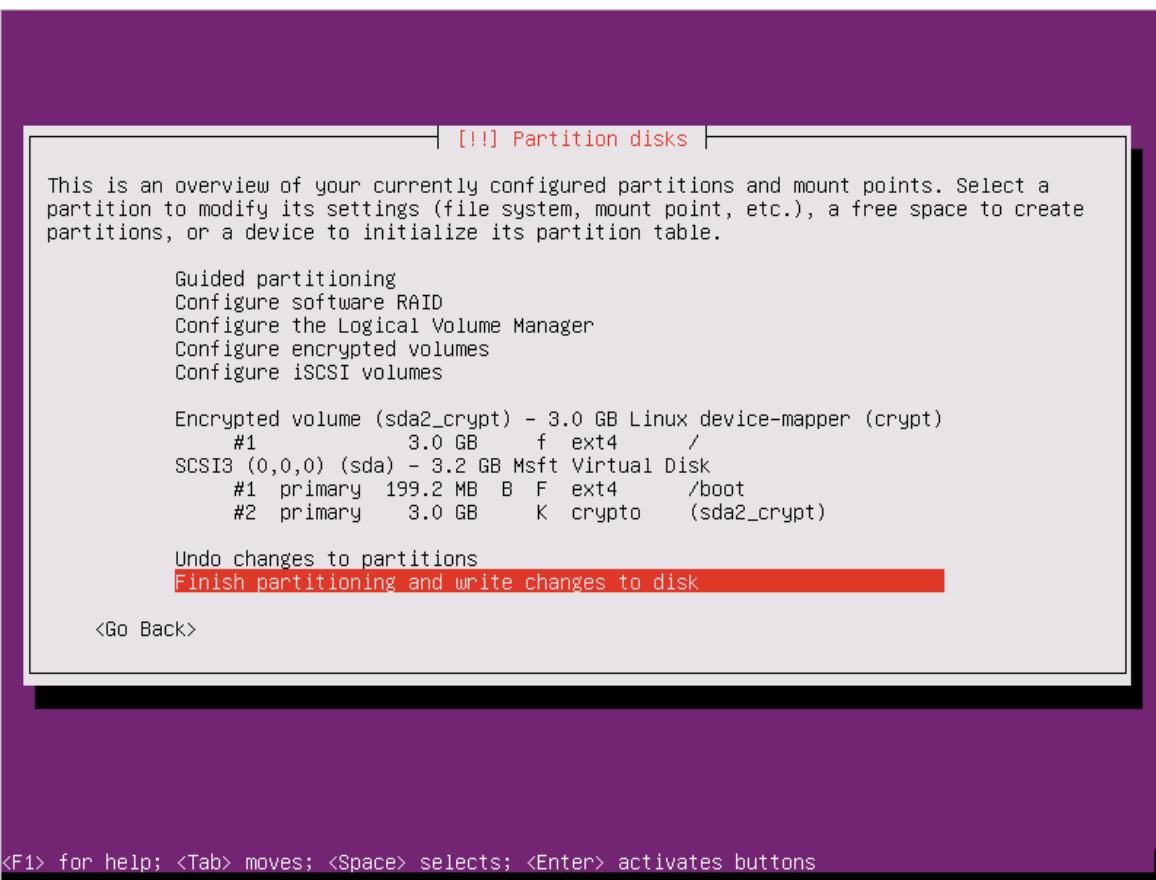


<Tab> moves; <Space> selects; <Enter> activates buttons

3. Provide a passphrase. This is the passphrase that you uploaded to the key vault.



4. Finish partitioning.



5. When you boot the VM and are asked for a passphrase, use the passphrase you provided in step 3.

```

[ 1.129797] input: Microsoft Vmbus HID-compliant Mouse as /devices/0006:045E:0621.0001/input/input4
[ 1.132206] sd: sda1 sda2
[ 1.133217] hid 0006:045E:0621.0001: input: <UNKNOWN> HID v0.01 Mouse [Microsoft Vmbus HID-compliant Mouse] on
[ 1.134340] hv_netvsc: hv_netvsc channel opened successfully
[ 1.138418] sd 2:0:0:0: [sda] Attached SCSI disk
[ 1.265049] hv_netvsc vmbus_15: Send section size: 6144, Section count:2560
[ 1.266137] hv_netvsc vmbus_15: Device MAC 00:15:5d:05:34:01 link state up
[ 1.272596] scsi host3: storvsc_host_t
[ 1.436076] psmouse serio1: trackpoint: failed to get extended button data
Begin: Loading essential drivers ... [ 2.401782] md: linear personality registered for level -1
[ 2.404316] md: multipath personality registered for level -4
[ 2.407122] md: raid0 personality registered for level 0
[ 2.410610] md: raid1 personality registered for level 1
[ 2.480009] raid6: sse2x1 gen() 10995 MB/s
[ 2.548012] raid6: sse2x1 xor() 8467 MB/s
[ 2.616010] raid6: sse2x2 gen() 14312 MB/s
[ 2.684013] raid6: sse2x2 xor() 9555 MB/s
[ 2.752011] raid6: sse2x4 gen() 16205 MB/s
[ 2.820010] raid6: sse2x4 xor() 11594 MB/s
[ 2.888007] raid6: avx2x1 gen() 21995 MB/s
[ 2.956007] raid6: avx2x2 gen() 25959 MB/s
[ 3.024011] raid6: avx2x4 gen() 29505 MB/s
[ 3.024735] raid6: using algorithm avx2x4 gen() 29505 MB/s
[ 3.025038] raid6: using avx2x2 recovery algorithm
[ 3.027102] xor: automatically using best checksumming function:
[ 3.064003] avx : 35013.000 MB/sec
[ 3.065688] async_tx: api initialized (async)
[ 3.074685] md: raid6 personality registered for level 6
[ 3.075435] md: raid5 personality registered for level 5
[ 3.075746] md: raid4 personality registered for level 4
[ 3.079565] md: raid10 personality registered for level 10
done.
Begin: Running /scripts/init-premount ... done.
Begin: Mounting root file system ... Begin: Running /scripts/local-top ... Please unlock disk sda2_crypt: _

```

6. Prepare the VM for uploading into Azure using [these instructions](#). Don't run the last step (deprovisioning the VM) yet.

Configure encryption to work with Azure by doing the following steps:

1. Create a file under /usr/local/sbin/azure\_crypt\_key.sh, with the content in the following script. Pay attention to the KeyFileName, because it's the passphrase file name used by Azure.

```

#!/bin/sh
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint
modprobe vfat >/dev/null 2>&1
modprobe ntfs >/dev/null 2>&1
sleep 2
OPENED=0
cd /sys/block
for DEV in sd*; do

    echo "> Trying device: $DEV ..." >&2
    mount -t vfat -r /dev/${DEV}1 $MountPoint >/dev/null ||
    mount -t ntfs -r /dev/${DEV}1 $MountPoint >/dev/null
    if [ -f $MountPoint/$KeyFileName ]; then
        cat $MountPoint/$KeyFileName
        umount $MountPoint 2>/dev/null
        OPENED=1
        break
    fi
    umount $MountPoint 2>/dev/null
done

if [ $OPENED -eq 0 ]; then
    echo "FAILED to find suitable passphrase file ..." >&2
    echo -n "Try to enter your password: " >&2
    read -s -r A </dev/console
    echo -n "$A"
else
    echo "Success loading keyfile!" >&2
fi

```

2. Change the crypt config in */etc/crypttab*. It should look like this:

```
xxx_crypt  uuid=xxxxxxxxxxxxxxxxxxxxxx none luks,discard,keyscript=/usr/local/sbin/azure_crypt_key.sh
```

3. Add executable permissions to the script:

```
chmod +x /usr/local/sbin/azure_crypt_key.sh
```

4. Edit */etc/initramfs-tools/modules* by appending lines:

```
vfat
ntfs
nls_cp437
nls_utf8
nls_iso8859-1
```

5. Run `update-initramfs -u -k all` to update the initramfs to make the `keyscript` take effect.

6. Now you can deprovision the VM.

```

root@ubuntu-preencrypted:~# ls -l /usr/local/sbin/azure_crypt_key.sh
-rwxr-xr-x 1 root root 860 Sep 18 16:57 /usr/local/sbin/azure_crypt_key.sh
root@ubuntu-preencrypted:~# cat /etc/crypttab
sda2_crypt UUID=b0dec704-1f2a-4f02-9a13-289c6c99dbb8 none luks,discard,keyscript=/usr/local/sbin/azure_crypt_key.sh
root@ubuntu-preencrypted:~# cat /etc/initramfs-tools/modules
# List of modules that you want to include in your initramfs.
# They will be loaded at boot time in the order below.

# Syntax: module_name [args ...]
# You must run update-initramfs(8) to effect this change.

# Examples:
#
# raid1
# sd_mod
# vfat
# ntfs
# nls_cp437
# nls_utf8
# nls_iso8859-1
root@ubuntu-preencrypted:~# update-initramfs -u -k all
update-initramfs: Generating /boot/initrd.img-4.4.0-36-generic
W: plymouth: The plugin label.so is missing, the selected theme might not work as expected.
W: plymouth: You might want to install the plymouth-themes and plymouth-label package to fix this.
W: mdadm: /etc/mdadm/mdadm.conf defines no arrays.
[ 6289.960173] blk_update_request: I/O error, dev fd0, sector 0
update-initramfs: Generating /boot/initrd.img-4.4.0-21-generic
W: plymouth: The plugin label.so is missing, the selected theme might not work as expected.
W: plymouth: You might want to install the plymouth-themes and plymouth-label package to fix this.
W: mdadm: /etc/mdadm/mdadm.conf defines no arrays.
[ 6297.592236] blk_update_request: I/O error, dev fd0, sector 0
root@ubuntu-preencrypted:~# waagent -force -deprovision
WARNING! The waagent service will be stopped.
WARNING! Cached DHCP leases will be deleted.
WARNING! root password will be disabled. You will not be able to login as root.
WARNING! Nameserver configuration in /etc/resolvconf/resolv.conf.d/{tail,original} will be deleted.
2016-09-18 17:06:38.572398 INFO resolvconf is enabled: leaving /etc/resolv.conf intact
2016-09-18 17:06:38.572398 INFO resolvconf is enabled: leaving /etc/resolv.conf intact
root@ubuntu-preencrypted:~# export HISTSIZE=0
root@ubuntu-preencrypted:~# logout

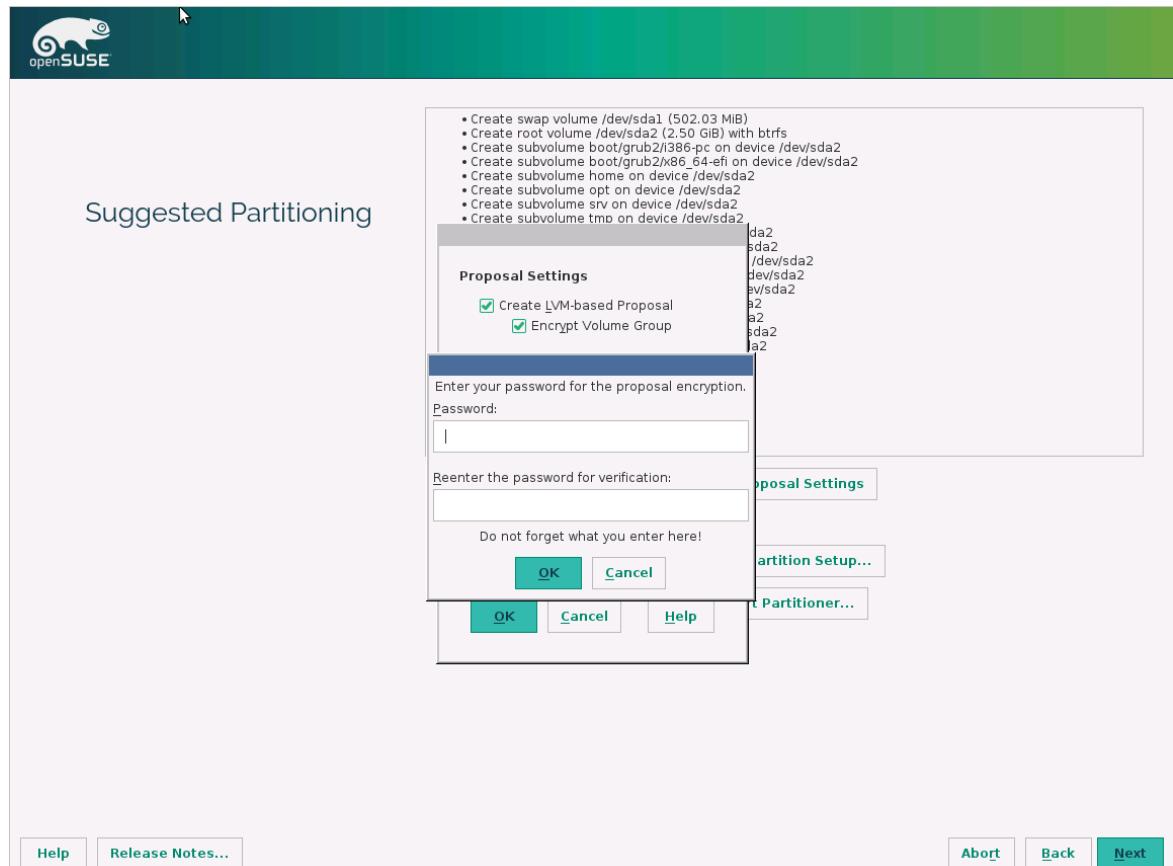
```

## 7. Continue to the next step and upload your VHD into Azure.

### openSUSE 13.2

To configure encryption during the distribution installation, do the following steps:

- When you partition the disks, select **Encrypt Volume Group**, and then enter a password. This is the password that you'll upload to your key vault.



2. Boot the VM using your password.

```
[ 0.000000] tsc: Fast TSC calibration failed
[ OK ] Found device Virtual_Disk.
[ OK ] Found device Virtual_Disk.
  Starting Cryptography Setup for cr_scsi-14d53465420202020fd10f64360...278fd632?ec-part2...
  Starting Setup Virtual Console...
[ OK ] Started Setup Virtual Console.
  Starting Dispatch Password Requests to Console...
[ OK ] Started Dispatch Password Requests to Console.
Please enter passphrase for disk Virtual_Disk (cr_scsi-14d53465420202020fd10f64360f5f14797052278fd632?ec-part2)! ****
```

3. Prepare the VM for uploading to Azure by following the instructions in [Prepare a SLES or openSUSE virtual machine for Azure](#). Don't run the last step (deprovisioning the VM) yet.

To configure encryption to work with Azure, do the following steps:

1. Edit the /etc/dracut.conf, and add the following line:

```
add_drivers+=" vfat ntfs nls_cp437 nls_iso8859-1"
```

2. Comment out these lines by the end of the file /usr/lib/dracut/modules.d/90crypt/module-setup.sh:

```
# inst_multiple -o \
#   $systemdutildir/system-generators/systemd-cryptsetup-generator \
#   $systemdutildir/systemd-cryptsetup \
#   $systemdsystemunitdir/systemd-ask-password-console.path \
#   $systemdsystemunitdir/systemd-ask-password-console.service \
#   $systemdsystemunitdir/cryptsetup.target \
#   $systemdsystemunitdir/sysinit.target.wants/cryptsetup.target \
#   # systemd-ask-password systemd-tty-ask-password-agent
#   inst_script "$moddir"/crypt-run-generator.sh /sbin/crypt-run-generator
```

3. Append the following line at the beginning of the file /usr/lib/dracut/modules.d/90crypt/parse-crypt.sh:

```
DRACUT_SYSTEMD=0
```

And change all occurrences of:

```
if [ -z "$DRACUT_SYSTEMD" ]; then
```

to:

```
if [ 1 ]; then
```

4. Edit `/usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh` and append it to "# Open LUKS device":

```
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint >&2
modprobe vfat >/dev/null >&2
modprobe ntfs >/dev/null >&2
for SFS in /dev/sd*; do
    echo "> Trying device:$SFS..." >&2
    mount ${SFS}1 $MountPoint -t vfat -r >&2 ||
    mount ${SFS}1 $MountPoint -t ntfs -r >&2
    if [ -f $MountPoint/$KeyFileName ]; then
        echo "> keyfile got..." >&2
        cp $MountPoint/$KeyFileName /tmp-keyfile >&2
        luksfile=/tmp-keyfile
        umount $MountPoint >&2
        break
    fi
done
```

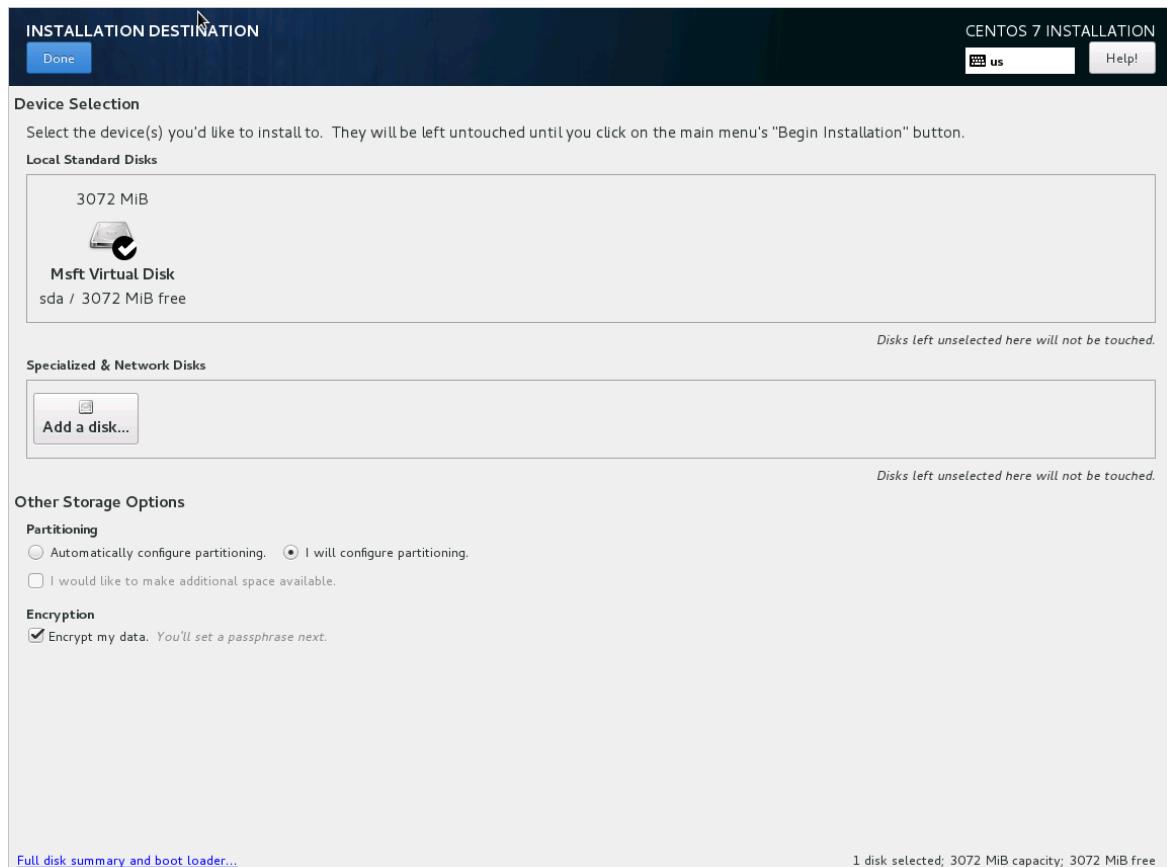
5. Run `/usr/sbin/dracut -f -v` to update the initrd.

6. Now you can deprovision the VM and upload your VHD into Azure.

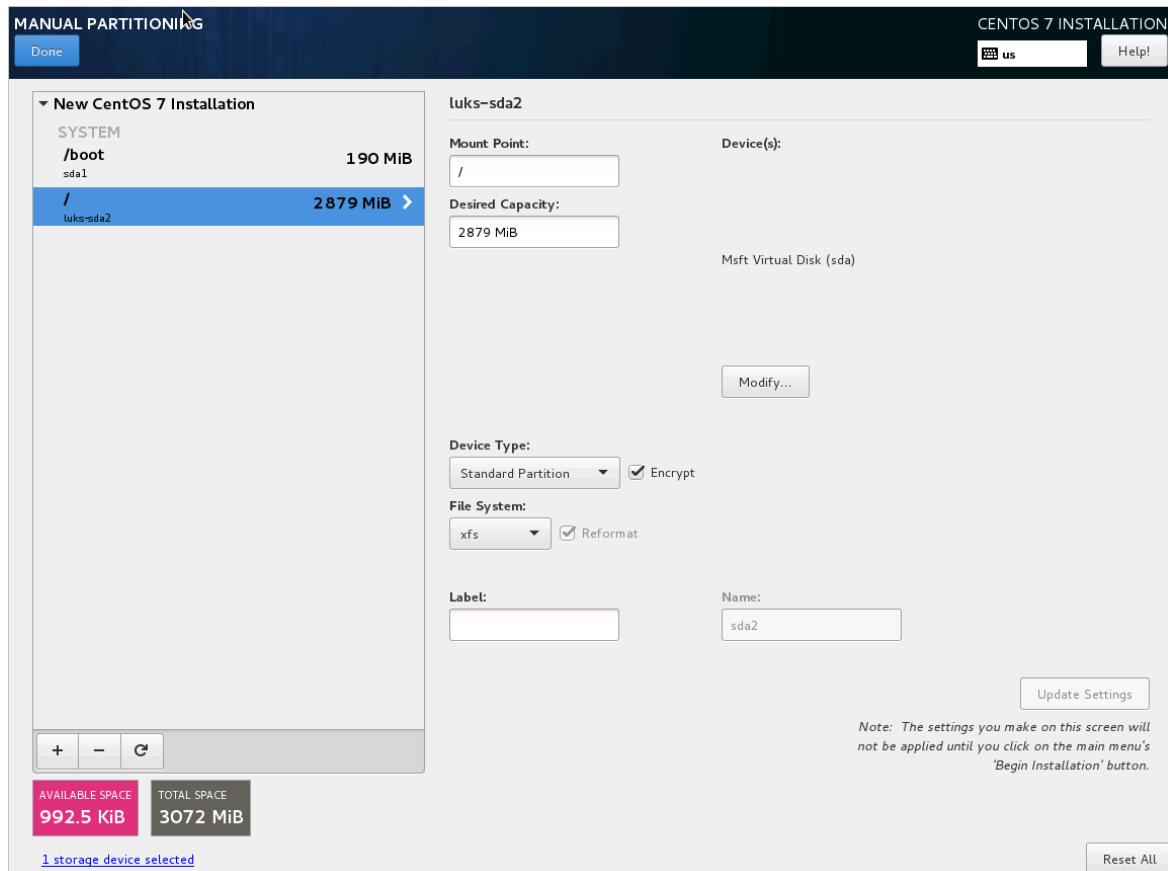
## CentOS 7 and RHEL 7

To configure encryption during the distribution installation, do the following steps:

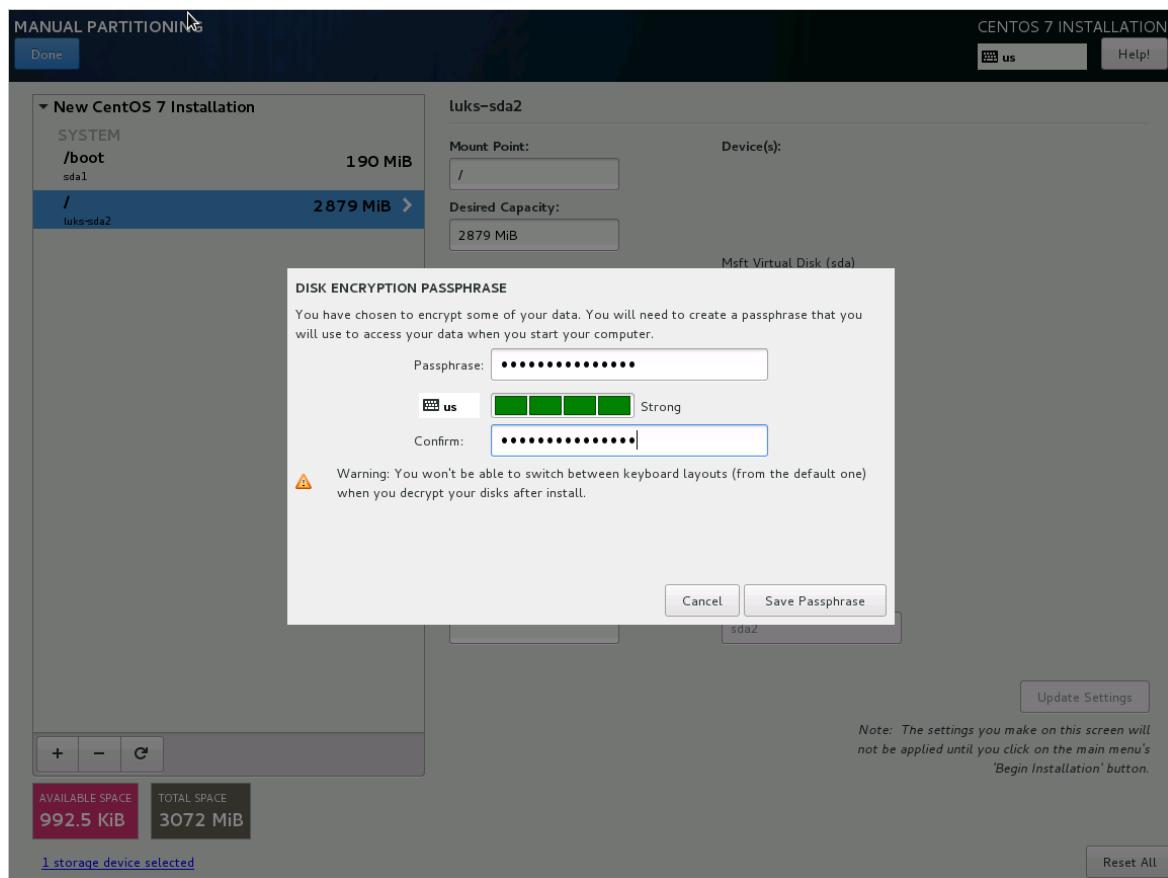
1. Select **Encrypt my data** when you partition disks.



2. Make sure Encrypt is selected for root partition.



3. Provide a passphrase. This is the passphrase that you'll upload to your key vault.



4. When you boot the VM and are asked for a passphrase, use the passphrase you provided in step 3.



```
Please enter passphrase for disk Virtual_Disk (luks-4e88b858-991e-4283-9ec5-130ec245cadf)!:*****
```

5. Prepare the VM for uploading into Azure by using the "CentOS 7.0+" instructions in [Prepare a CentOS-based virtual machine for Azure](#). Don't run the last step (deprovisioning the VM) yet.

6. Now you can deprovision the VM and upload your VHD into Azure.

To configure encryption to work with Azure, do the following steps:

1. Edit the /etc/dracut.conf, and add the following line:

```
add_drivers+=" vfat ntfs nls_cp437 nls_iso8859-1"
```

2. Comment out these lines by the end of the file /usr/lib/dracut/modules.d/90crypt/module-setup.sh:

```
# inst_multiple -o \
# $systemdutildir/system-generators/systemd-cryptsetup-generator \
# $systemdutildir/systemd-cryptsetup \
# $systemdsystemunitdir/systemd-ask-password-console.path \
# $systemdsystemunitdir/systemd-ask-password-console.service \
# $systemdsystemunitdir/cryptsetup.target \
# $systemdsystemunitdir/sysinit.target.wants/cryptsetup.target \
# systemd-ask-password systemd-tty-ask-password-agent
# inst_script "$moddir"/crypt-run-generator.sh /sbin/crypt-run-generator
```

3. Append the following line at the beginning of the file /usr/lib/dracut/modules.d/90crypt/parse-crypt.sh:

```
DRACUT_SYSTEMD=0
```

And change all occurrences of:

```
if [ -z "$DRACUT_SYSTEMD" ]; then
```

to

```
if [ 1 ]; then
```

4. Edit /usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh and append the following after the "# Open LUKS device":

```
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint >&2
modprobe vfat >/dev/null >&2
modprobe ntfs >/dev/null >&2
for SFS in /dev/sd*; do
echo "> Trying device:$SFS..." >&2
mount ${SFS}1 $MountPoint -t vfat -r >&2 ||
mount ${SFS}1 $MountPoint -t ntfs -r >&2
if [ -f $MountPoint/$KeyFileName ]; then
    echo "> keyfile got..." >&2
    cp $MountPoint/$KeyFileName /tmp-keyfile >&2
    luksfile=/tmp-keyfile
    umount $MountPoint >&2
    break
fi
done
```

5. Run the "/usr/sbin/dracut -f -v" to update the initrd.

```
[root@centos-preencrypted ~]# cat /etc/dracut.conf | grep add_drivers
add_drivers+=" vfat ntfs nls_cp437 nls_iso8859-1"
[root@centos-preencrypted ~]# cat /usr/lib/dracut/modules.d/90crypt/cryptroot-ask.sh | grep LinuxPassPhraseFileName -A 15 -B 1
MountPoint=/tmp-keydisk-mount
KeyFileName=LinuxPassPhraseFileName
echo "Trying to get the key from disks ..." >&2
mkdir -p $MountPoint >&2
modprobe vfat >/dev/null >&2
modprobe ntfs >/dev/null >&2
for SFS in /dev/sd*; do
echo "> Trying device:$SFS..." >&2
mount ${SFS}1 $MountPoint -t vfat -r >&2 ||
mount ${SFS}1 $MountPoint -t ntfs -r >&2
if [ -f $MountPoint/$KeyFileName ]; then
    echo "> keyfile got..." >&2
    cp $MountPoint/$KeyFileName /tmp-keyfile >&2
    luksfile=/tmp-keyfile
    umount $MountPoint >&2
    break
fi
[root@centos-preencrypted ~]# dracut -f -v_
```

## Upload encrypted VHD to an Azure storage account

After DM-Crypt encryption is enabled, the local encrypted VHD needs to be uploaded to your storage account.

```
Add-AzVhd [-Destination] <Uri> [-LocalFilePath] <FileInfo> [[-NumberOfUploaderThreads] <Int32> ] [[-BaseImageUriToPatch] <Uri> ] [[-OverWrite]] [ <CommonParameters>]
```

## Upload the secret for the pre-encrypted VM to your key vault

When encrypting using an Azure AD app (previous release), the disk-encryption secret that you obtained previously must be uploaded as a secret in your key vault. The key vault needs to have disk encryption and permissions enabled for your Azure AD client.

```
$AadClientId = "My-AAD-Client-Id"
$AadClientSecret = "My-AAD-Client-Secret"

$keyVault = New-AzKeyVault -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -Location $Location

Set-AzKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -ServicePrincipalName $AadClientId -PermissionsToKeys all -PermissionsToSecrets all
Set-AzKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -EnabledForDiskEncryption
```

### Disk encryption secret not encrypted with a KEK

To set up the secret in your key vault, use [Set-AzKeyVaultSecret](#). The passphrase is encoded as a base64 string and then uploaded to the key vault. In addition, make sure that the following tags are set when you create the secret in the key vault.

```
# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

$tags = @{"DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP"; "DiskEncryptionKeyFileName" = "LinuxPassPhraseFileName"}
$secretName = [guid]::NewGuid().ToString()
$secretValue = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($passphrase))
$secureSecretValue = ConvertTo-SecureString $secretValue -AsPlainText -Force

$secret = Set-AzKeyVaultSecret -VaultName $KeyVaultName -Name $secretName -SecretValue $secureSecretValue -tags $tags
$secretUrl = $secret.Id
```

Use the `$secretUrl` in the next step for [attaching the OS disk without using KEK](#).

### Disk encryption secret encrypted with a KEK

Before you upload the secret to the key vault, you can optionally encrypt it by using a key encryption key. Use the wrap API to first encrypt the secret using the key encryption key. The output of this wrap operation is a base64 URL encoded string, which you can then upload as a secret by using the [Set-AzKeyVaultSecret](#) cmdlet.

```
# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

Add-AzKeyVaultKey -VaultName $KeyVaultName -Name "keyencryptionkey" -Destination Software
$keyEncryptionKey = Get-AzKeyVaultKey -VaultName $KeyVault.OriginalVault.Name -Name "keyencryptionkey"

$apiversion = "2015-06-01"

#####
# Get Auth URI
#####
```

```

$uri = $KeyVault.VaultUri + "/keys"
$headers = @{}

$response = try { Invoke-RestMethod -Method GET -Uri $uri -Headers $headers } catch {
$_._Exception.Response }

$authHeader = $response.Headers["www-authenticate"]
$authUri = [regex]::match($authHeader, 'authorization="(.*)"').Groups[1].Value

Write-Host "Got Auth URI successfully"

#####
# Get Auth Token
#####

$uri = $authUri + "/oauth2/token"
$body = "grant_type=client_credentials"
$body += "&client_id=" + $AadClientId
$body += "&client_secret=" + [Uri]::EscapeDataString($AadClientSecret)
$body += "&resource=" + [Uri]::EscapeDataString("https://vault.azure.net")
$headers = @{}

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$access_token = $response.access_token

Write-Host "Got Auth Token successfully"

#####
# Get KEK info
#####

$uri = $KeyEncryptionKey.Id + "?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token}

$response = Invoke-RestMethod -Method GET -Uri $uri -Headers $headers

$keyid = $response.key.kid

Write-Host "Got KEK info successfully"

#####
# Encrypt passphrase using KEK
#####

$passphraseB64 = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Passphrase))
$uri = $keyid + "/encrypt?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"alg" = "RSA-OAEP"; "value" = $passphraseB64}
$body = $bodyObj | ConvertTo-Json

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$wrappedSecret = $response.value

Write-Host "Encrypted passphrase successfully"

#####
# Store secret
#####

$secretName = [guid]::NewGuid().ToString()
$uri = $KeyVault.VaultUri + "/secrets/" + $secretName + "?api-version=" + $apiversion
$secretAttributes = @{"enabled" = $true}
$secretTags = @{"DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP"; "DiskEncryptionKeyFileName" =
"LinuxPassPhraseFileName"}
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"value" = $wrappedSecret; "attributes" = $secretAttributes; "tags" = $secretTags}
$body = $bodyObj | ConvertTo-Json

```

```
$response = Invoke-RestMethod -Method PUT -Uri $uri -Headers $headers -Body $body  
  
Write-Host "Stored secret successfully"  
  
$secretUrl = $response.id
```

Use `$KeyEncryptionKey` and `$secretUrl` in the next step for [attaching the OS disk using KEK](#).

## Specify a secret URL when you attach an OS disk

### Without using a KEK

While you're attaching the OS disk, you need to pass `$secretUrl`. The URL was generated in the "Disk-encryption secret not encrypted with a KEK" section.

```
Set-AzVMOSDisk `  
    -VM $VirtualMachine `  
    -Name $OSDiskName `  
    -SourceImageUri $VhdUri `  
    -VhdUri $OSDiskUri `  
    -Linux `  
    -CreateOption FromImage `  
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId `  
    -DiskEncryptionKeyUrl $SecretUrl
```

### Using a KEK

When you attach the OS disk, pass `$KeyEncryptionKey` and `$secretUrl`. The URL was generated in the "Disk-encryption secret encrypted with a KEK" section.

```
Set-AzVMOSDisk `  
    -VM $VirtualMachine `  
    -Name $OSDiskName `  
    -SourceImageUri $CopiedTemplateBlobUri `  
    -VhdUri $OSDiskUri `  
    -Linux `  
    -CreateOption FromImage `  
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId `  
    -DiskEncryptionKeyUrl $SecretUrl `  
    -KeyEncryptionKeyVaultId $KeyVault.ResourceId `  
    -KeyEncryptionKeyURL $KeyEncryptionKey.Id
```

# Azure Disk Encryption on an isolated network

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When connectivity is restricted by a firewall, proxy requirement, or network security group (NSG) settings, the ability of the extension to perform needed tasks might be disrupted. This disruption can result in status messages such as "Extension status not available on the VM."

## Package management

Azure Disk Encryption depends on a number of components, which are typically installed as part of ADE enablement if not already present. When behind a firewall or otherwise isolated from the Internet, these packages must be pre-installed or available locally.

Here are the packages necessary for each distribution. For a full list of supported distros and volume types, see [supported VMs and operating systems](#).

- **Ubuntu 14.04, 16.04, 18.04:** lsscsi, psmisc, at, cryptsetup-bin, python-parted, python-six, procps, grub-pc-bin
- **CentOS 7.2 - 7.9, 8.1, 8.2:** lsscsi, psmisc, lvm2, uuid, at, patch, cryptsetup, cryptsetup-reencrypt, pyparted, procps-ng, util-linux
- **CentOS 6.8:** lsscsi, psmisc, lvm2, uuid, at, cryptsetup-reencrypt, pyparted, python-six
- **RedHat 7.2 - 7.9, 8.1, 8.2:** lsscsi, psmisc, lvm2, uuid, at, patch, cryptsetup, cryptsetup-reencrypt, procps-ng, util-linux
- **RedHat 6.8:** lsscsi, psmisc, lvm2, uuid, at, patch, cryptsetup-reencrypt
- **openSUSE 42.3, SLES 12-SP4, 12-SP3:** lsscsi, cryptsetup

On Red Hat, when a proxy is required, you must make sure that the subscription-manager and yum are set up properly. For more information, see [How to troubleshoot subscription-manager and yum problems](#).

When packages are installed manually, they must also be manually upgraded as new versions are released.

## Network security groups

Any network security group settings that are applied must still allow the endpoint to meet the documented network configuration prerequisites for disk encryption. See [Azure Disk Encryption: Networking requirements](#)

## Azure Disk Encryption with Azure AD (previous version)

If using [Azure Disk Encryption with Azure AD \(previous version\)](#), the [Azure Active Directory Library](#) will need to be installed manually for all distros (in addition to the packages appropriate for the distro, as [listed above](#)).

When encryption is being enabled with [Azure AD credentials](#), the target VM must allow connectivity to both Azure Active Directory endpoints and Key Vault endpoints. Current Azure Active Directory authentication endpoints are maintained in sections 56 and 59 of the [Microsoft 365 URLs and IP address ranges](#) documentation. Key Vault instructions are provided in the documentation on how to [Access Azure Key Vault behind a firewall](#).

### Azure Instance Metadata Service

The virtual machine must be able to access the [Azure Instance Metadata service](#) endpoint, which uses a well-

known non-routable IP address ( `169.254.169.254` ) that can be accessed only from within the VM. Proxy configurations that alter local HTTP traffic to this address (for example, adding an X-Forwarded-For header) are not supported.

## Next steps

- See more steps for [Azure disk encryption troubleshooting](#)
- [Azure data encryption at rest](#)

# Verify encryption status for Linux

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The scope of this article is to validate the encryption status of a virtual machine by using different methods: the Azure portal, PowerShell, the Azure CLI, or the operating system of the virtual machine (VM).

You can validate the encryption status during or after the encryption, by either:

- Checking the disks attached to a particular VM.
- Querying the encryption settings on each disk, whether the disk is attached or unattached.

This scenario applies for Azure Disk Encryption dual-pass and single-pass extensions. Linux distributions are the only environment for this scenario.

## NOTE

We're using variables throughout the article. Replace the values accordingly.

## Portal

In the Azure portal, inside the **Extensions** section, select the Azure Disk Encryption extension in the list. The information for **Status message** indicates the current encryption status:

A screenshot of the Azure portal showing the Azure Disk Encryption extension details. The extension name is 'AzureDiskEncryptionForLinux' and it is associated with the resource group 'vmadebackup'. The 'Status' field shows 'Provisioning succeeded'. The 'Version' field shows '1.1.0.48'. The 'Status message' field shows 'Install Succeeded'. The 'Detailed status' link is visible. The 'Handler status' is 'Ready' and the 'Handler status level' is 'Info'. The 'Resource ID' is a long URL starting with '/subscriptions/...'. The 'Status' and 'Version' fields are highlighted with red boxes.

Type	Microsoft.Azure.Security.AzureDiskEncryptionForLinux
Status	Provisioning succeeded
Version	1.1.0.48
Status level	Info
Status message	Install Succeeded
Detailed status	<a href="#">View detailed status</a>
Handler status	Ready
Handler status level	Info
Resource ID	/subscriptions/<subscription id>/

In the list of extensions, you'll see the corresponding Azure Disk Encryption extension version. Version 0.x corresponds to Azure Disk Encryption dual pass, and version 1.x corresponds to Azure Disk Encryption single pass.

You can get more details by selecting the extension and then selecting **View detailed status**. The detailed status of the encryption process appears in JSON format.

## AzureDiskEncryptionForLinux

vmadebackup

<a href="#">Uninstall</a>	
Type	Microsoft.Azure.Security.AzureDiskEncryptionForLinux
Status	Provisioning succeeded
Version	1.1.0.48
Status level	Info
Status message	Install Succeeded
Detailed status	<a href="#">View detailed status</a>
Handler status	Ready
Handler status level	Info
Resource ID	/subscriptions/ <subscription id> /

## AzureDiskEncryptionForLinux

vmadebackup

```
1 [ {  
2   "code": "ComponentStatus/Microsoft.Azure.Security.Azu  
3   "level": "Info",  
4   "displayStatus": "Provisioning succeeded",  
5   "message": "{\"os\": \"Encrypted\", \"data\": \"Encry  
6 }  
7 ]  
8 }
```

Another way to validate the encryption status is by looking at the **Disk settings** section.

Disk settings

Enable Ultra Disk compatibility  Yes  No

**OS disk**

Name	Size	Storage account ...	Encryption	Host caching
vmadebackup_OsDisk_1_7ffb0997763c483...	32 GiB	Premium SSD	<b>Enabled</b>	Read/write

**Data disks**

LUN	Name	Size	Storage account ...	Encryption	Host caching
0	vmadebackup_datadisk0	5 GiB	Standard HDD	<b>Enabled</b>	None

### NOTE

This status means the disks have encryption settings stamped, not that they were actually encrypted at the OS level.

By design, the disks are stamped first and encrypted later. If the encryption process fails, the disks may end up stamped but not encrypted.

To confirm if the disks are truly encrypted, you can double check the encryption of each disk at the OS level.

## PowerShell

You can validate the *general* encryption status of an encrypted VM by using the following PowerShell commands:

```
$VMNAME="VMNAME"
$RGNAME="RGNAME"
Get-AzVmDiskEncryptionStatus -ResourceGroupName ${RGNAME} -VMName ${VMNAME}
```

```
PS Azure:\> Get-AzVmDiskEncryptionStatus -ResourceGroupName ${RGNAME} -VMName ${VMNAME}

OsVolumeEncrypted      : NotEncrypted
DataVolumesEncrypted   : Encrypted
OsVolumeEncryptionSettings :
ProgressMessage        : Provisioning succeeded
```

You can capture the encryption settings from each disk by using the following PowerShell commands.

### Single pass

In a single pass, the encryption settings are stamped on each of the disks (OS and data). You can capture the encryption settings for an OS disk in a single pass as follows:

```
$RGNAME = "RGNAME"
$VMNAME = "VMNAME"

$VM = Get-AzVM -Name ${VMNAME} -ResourceGroupName ${RGNAME}
$Sourcedisk = Get-AzDisk -ResourceGroupName ${RGNAME} -DiskName $VM.StorageProfile.OsDisk.Name
Write-Host
=====
=====
Write-Host "Encryption Settings:"
Write-Host
=====
=====
Write-Host "Enabled:" $Sourcedisk.EncryptionSettingsCollection.Enabled
Write-Host "Version:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettingsVersion
Write-Host "Source Vault:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SourceVault.Id
Write-Host "Secret URL:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SecretUrl
Write-Host "Key URL:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.KeyEncryptionKey.KeyUrl
Write-Host
=====
=====
```

```
===== Encryption Settings:
=====
Enabled: True
Version: 1.0
Source Vault: /subscriptions/ <subscription id> /resourceGroups/rgadabackup/providers/Microsoft.KeyVault/vaults/kvdualadebackup
Secret URL: https://kvdualadebackup.vault.azure.net/secrets/fa721375-4682-42e3-9ab3-bfe3f6ec8e4b/4f60f2eac6fa492f8272593030c4bbb9
Key URL: https://kvdualadebackup.vault.azure.net/keys/keydualadebackup/d656f60357fb491e902961a1eea6409c
=====
```

If the disk doesn't have encryption settings stamped, the output will be empty:

```
lvmincreaseremovedataenc_OsDisk_1_a73e76483c144d4886c217df200b2f3a
Encryption Settings:
PS C:\WINDOWS\system32> |
```

Use the following commands to capture encryption settings for data disks:

```

$RGNAME = "RGNAME"
$VMNAME = "VMNAME"

$VM = Get-AzVM -Name ${VMNAME} -ResourceGroupName ${RGNAME}
clear
foreach ($i in $VM.StorageProfile.DataDisks|ForEach-Object{$__.Name})
{
Write-Host
"=====
=====
Write-Host "Encryption Settings:"
Write-Host
"=====
=====
Write-Host "Checking Disk:" $i
$Disk=(Get-AzDisk -ResourceGroupName ${RGNAME} -DiskName $i)
Write-Host "Encryption Enable: " $Sourcedisk.EncryptionSettingsCollection.Enabled
Write-Host "Encryption KeyEncryptionKey: "
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.KeyEncryptionKey.KeyUrl;
Write-Host "Encryption DiskEncryptionKey: "
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SecretUrl;
Write-Host
"=====
=====
}

```

```

Encryption Settings:
=====
Checking Disk: vmdual_datadisk0
Encryption Enable: True
Encryption KeyEncryptionKey: https://kvdualebackup.vault.azure.net/keys/kvdualebackup/d656f60357fb491e902961a1eea6409c
Encryption DiskEncryptionKey: https://kvdualebackup.vault.azure.net/secrets/fa721375-4682-42e3-9ab3-bfe3f6ec8e4b/4f60f2eac6fa492f8272593030c4bbb9
=====
```

## Dual pass

In a dual pass, the encryption settings are stamped in the VM model and not on each individual disk.

To verify that the encryption settings were stamped in a dual pass, use the following commands:

```

$RGNAME = "RGNAME"
$VMNAME = "VMNAME"

$vm = Get-AzVm -ResourceGroupName ${RGNAME} -Name ${VMNAME};
$Sourcedisk = Get-AzDisk -ResourceGroupName ${RGNAME} -DiskName $VM.StorageProfile.OsDisk.Name
clear
Write-Host
"=====
=====
Write-Host "Encryption Settings:"
Write-Host
"=====
=====
Write-Host "Enabled:" $Sourcedisk.EncryptionSettingsCollection.Enabled
Write-Host "Version:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettingsVersion
Write-Host "Source Vault:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SourceVault.Id
Write-Host "Secret URL:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SecretUrl
Write-Host "Key URL:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.KeyEncryptionKey.KeyUrl
Write-Host
"=====
=====
```

```

Encryption Settings:
=====
Enabled: True
Version: 1.0
Source Vault: /subscriptions/<subscription_id>/resourceGroups/rгадебакуп/providers/Microsoft.KeyVault/vaults/kvdualebackup
Secret URL: https://kvdualebackup.vault.azure.net/secrets/fa721375-4682-42e3-9ab3-bfe3f6ec8e4b/4f60f2eac6fa492f8272593030c4bbb9
Key URL: https://kvdualebackup.vault.azure.net/keys/kvdualebackup/d656f60357fb491e902961a1eea6409c
=====
```

## Unattached disks

Check the encryption settings for disks that aren't attached to a VM.

## Managed disks

```
$Sourcedisk = Get-AzDisk -ResourceGroupName ${RGNAME} -DiskName ${TARGETDISKNAME}
Write-Host
=====
=====
Write-Host "Encryption Settings:"
Write-Host
=====
=====
Write-Host "Enabled:" $Sourcedisk.EncryptionSettingsCollection.Enabled
Write-Host "Version:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettingsVersion
Write-Host "Source Vault:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SourceVault.Id
Write-Host "Secret URL:"
$Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.DiskEncryptionKey.SecretUrl
Write-Host "Key URL:" $Sourcedisk.EncryptionSettingsCollection.EncryptionSettings.KeyEncryptionKey.KeyUrl
Write-Host
=====
=====
```

## Azure CLI

You can validate the *general*/encryption status of an encrypted VM by using the following Azure CLI commands:

```
VMNAME="VMNAME"
RGNAME="RGNAME"
az vm encryption show --name ${VMNAME} --resource-group ${RGNAME} --query "substatus"
```

```
Code          $ az vm encryption show --name ${VMNAME} --resource-group ${RGNAME} --query "substatus" -o table
              Level   DisplayStatus      Message
----- -----
ComponentStatus/Microsoft.Azure.Security.AzureDiskEncryptionForLinux/succeeded  Info    Provisioning succeeded {"os": "Encrypted", "data": "Encrypted"}
```

### Single pass

You can validate the encryption settings for each disk by using the following Azure CLI commands:

```
az vm encryption show -g ${RGNAME} -n ${VMNAME} --query "disks[*].[name, statuses[*].displayStatus]" -o table
```

Column1	Column2
lvmincreaseremovedataenc_OsDisk_1_a73e76483c144d4886c217df200b2f3a	['Disk is not encrypted']
lvmincreaseremovedataenc_datadisk1	['Encryption is enabled on disk']
lvmincreaseremovedataenc_datadisk2	['Encryption is enabled on disk']
lvmincreaseremovedataenc_datadisk3	['Encryption is enabled on disk']
lvmincreaseremovedataenc_datadisk4	['Encryption is enabled on disk']

### IMPORTANT

If the disk doesn't have encryption settings stamped, you'll see the text **Disk is not encrypted**.

Use the following commands to get detailed status and encryption settings.

OS disk:

```

RGNAME="RGNAME"
VMNAME="VNAME"

disk=`az vm show -g ${RGNAME} -n ${VMNAME} --query storageProfile.osDisk.name -o tsv`  

for disk in $disk; do \  

echo  

=====
=====  

echo -ne "Disk Name: "; az disk show -g ${RGNAME} -n ${disk} --query name -o tsv; \  

echo -ne "Encryption Enabled: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.enabled -o tsv; \  

echo -ne "Version: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  

encryptionSettingsCollection.encryptionSettingsVersion -o tsv; \  

echo -ne "Disk Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.encryptionSettings[].diskEncryptionKey.secretUrl -o tsv; \  

echo -ne "key Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.encryptionSettings[].keyEncryptionKey.keyUrl -o tsv; \  

echo  

=====
=====  

done

```

```

=====
Disk Name: vmadebackup_OsDisk_1_7ffb0997763c483881d53c54e58f4898
Encryption Enabled: true
Disk Encryption Key: https://kvadebackup.vault.azure.net/secrets/5dfe7d9d-c77e-478f-aa2d-321f62c394d9/6588ab5e10f54423b0f35ec6a3a64f1d
key Encryption Key: https://kvadebackup.vault.azure.net/keys/keyadebackup/1551be9b670947428b04b6aa0a31a040
=====
```

Data disks:

```

RGNAME="RGNAME"
VMNAME="VNAME"
az vm encryption show --name ${VMNAME} --resource-group ${RGNAME} --query "substatus"

for disk in `az vm show -g ${RGNAME} -n ${VMNAME} --query storageProfile.dataDisks[].name -o tsv`; do \  

echo  

=====
=====  

echo -ne "Disk Name: "; az disk show -g ${RGNAME} -n ${disk} --query name -o tsv; \  

echo -ne "Encryption Enabled: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.enabled -o tsv; \  

echo -ne "Version: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  

encryptionSettingsCollection.encryptionSettingsVersion -o tsv; \  

echo -ne "Disk Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.encryptionSettings[].diskEncryptionKey.secretUrl -o tsv; \  

echo -ne "key Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  

encryptionSettingsCollection.encryptionSettings[].keyEncryptionKey.keyUrl -o tsv; \  

echo  

=====
=====  

done

```

```

=====
Disk Name: vmadebackup_datadisk0
Encryption Enabled: true
Disk Encryption Key: https://kvadebackup.vault.azure.net/secrets/5dfe7d9d-c77e-478f-aa2d-321f62c394d9/4d0c7805d33e47dd81838aabccfff85a
key Encryption Key: https://kvadebackup.vault.azure.net/keys/keyadebackup/1551be9b670947428b04b6aa0a31a040
=====
```

## Dual pass

```
az vm encryption show --name ${VMNAME} --resource-group ${RGNAME} -o table
```

OsDisk	DataDisk	OsType	ProgressMessage
Encrypted	Encrypted	Linux	https://kv dual a backup.vault.azure.net/secrets/fa721375-4682-42e3-9ab3-bfe3f6ec8e4b/4f60f2eac6fa492f8272593030c4bb9

You can also check the encryption settings on the VM Model Storage profile of the OS disk:

```
disk=`az vm show -g ${RGNAME} -n ${VMNAME} --query storageProfile.osDisk.name -o tsv`  
for disk in $disk; do \  
echo  
"=====  
====="; \  
echo -ne "Disk Name: "; az disk show -g ${RGNAME} -n ${disk} --query name -o tsv; \  
echo -ne "Encryption Enabled: "; az disk show -g ${RGNAME} -n ${disk} --query  
encryptionSettingsCollection.enabled -o tsv; \  
echo -ne "Version: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  
encryptionSettingsCollection.encryptionSettingsVersion -o tsv; \  
echo -ne "Disk Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  
encryptionSettingsCollection.encryptionSettings[].diskEncryptionKey.secretUrl -o tsv; \  
echo -ne "key Encryption Key: "; az disk show -g ${RGNAME} -n ${disk} --query  
encryptionSettingsCollection.encryptionSettings[].keyEncryptionKey.keyUrl -o tsv; \  
echo  
"=====  
====="  
done
```

```
=====  
Disk Name: vmdual_OsDisk_1_16c83fa21fe649a8846d395a9104bd2b  
Encryption Enabled: true  
Disk Encryption Key: https://kvdualebackup.vault.azure.net/secrets/fa721375-4682-42e3-9ab3-bfe3f6ec8e4b/4f60f2eac6fa492f8272593030c4bbb9  
key Encryption Key: https://kvdualebackup.vault.azure.net/keys/keydualebackup/d656f60357fb491e902961a1eea6409c  
=====
```

## Unattached disks

Check the encryption settings for disks that aren't attached to a VM.

## Managed disks

```
RGNAME="RGNAME"  
TARGETDISKNAME="DISKNAME"  
echo  
"=====  
====="; \  
echo -ne "Disk Name: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query name -o tsv; \  
echo -ne "Encryption Enabled: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  
encryptionSettingsCollection.enabled -o tsv; \  
echo -ne "Version: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  
encryptionSettingsCollection.encryptionSettingsVersion -o tsv; \  
echo -ne "Disk Encryption Key: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  
encryptionSettingsCollection.encryptionSettings[].diskEncryptionKey.secretUrl -o tsv; \  
echo -ne "key Encryption Key: "; az disk show -g ${RGNAME} -n ${TARGETDISKNAME} --query  
encryptionSettingsCollection.encryptionSettings[].keyEncryptionKey.keyUrl -o tsv; \  
echo  
"=====  
====="
```

## Unmanaged disks

Unmanaged disks are VHD files that are stored as page blobs in Azure storage accounts.

To get the details for a specific disk, you need to provide:

- The ID of the storage account that contains the disk.
- A connection string for that particular storage account.
- The name of the container that stores the disk.
- The disk name.

This command lists all the IDs for all your storage accounts:

```
az storage account list --query [].[id] -o tsv
```

The storage account IDs are listed in the following form:

```
/subscriptions/<subscription id>/resourceGroups/<resource group  
name>/providers/Microsoft.Storage/storageAccounts/<storage account name>
```

Select the appropriate ID and store it on a variable:

```
id="/subscriptions/<subscription id>/resourceGroups/<resource group  
name>/providers/Microsoft.Storage/storageAccounts/<storage account name>"
```

This command gets the connection string for one particular storage account and stores it on a variable:

```
ConnectionString=$(az storage account show-connection-string --ids $id --query connectionString -o tsv)
```

The following command lists all the containers under a storage account:

```
az storage container list --connection-string $ConnectionString --query [].[name] -o tsv
```

The container used for disks is normally named "vhds."

Store the container name on a variable:

```
ContainerName="name of the container"
```

Use this command to list all the blobs on a particular container:

```
az storage blob list -c ${ContainerName} --connection-string $ConnectionString --query [].[name] -o tsv
```

Choose the disk that you want to query and store its name on a variable:

```
DiskName="diskname.vhd"
```

Query the disk encryption settings:

```
az storage blob show -c ${ContainerName} --connection-string ${ConnectionString} -n ${DiskName} --query  
metadata.DiskEncryptionSettings
```

## Operating system

Validate if the data disk partitions are encrypted (and the OS disk isn't).

When a partition or disk is encrypted, it's displayed as a **crypt** type. When it's not encrypted, it's displayed as a **part/disk** type.

```
lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot
└─sda2	8:2	0	31.5G	0	part	
└─osencrypt	253:0	0	31.5G	0	crypt	/
sdb	8:16	0	32G	0	disk	
└─sdb1	8:17	0	32G	0	part	/mnt/resource
sdc	8:32	0	48M	0	disk	
└─sdc1	8:33	0	47M	0	part	/mnt/azure_bek_disk
sdd	8:48	0	5G	0	disk	
└─sdd1	8:49	0	5G	0	part	
└─myencrypteddisk	253:1	0	5G	0	crypt	/data4tb

You can get more details by using the following `lsblk` variant.

You'll see a `crypt` type layer that is mounted by the extension. The following example shows logical volumes and normal disks having `crypto_LUKS` FSTYPE.

```
lsblk -o NAME,TYPE,FSTYPE,LABEL,SIZE,RO,MOUNTPOINT
```

NAME	TYPE	FSTYPE	LABEL	SIZE	RO	MOUNTPOINT
fd0	disk			4K	0	
sda	disk			32G	0	
└─sda1	part	xfs		500M	0	/boot
└─sda2	part			31.5G	0	
└─osencrypt	crypt	xfs		31.5G	0	/
sdb	disk			32G	0	
└─sdb1	part	ext4		32G	0	/mnt/resource
sdc	disk			48M	0	
└─sdc1	part	vfat	BEK VOLUME	47M	0	/mnt/azure_bek_disk
sdd	disk			5G	0	
└─sdd1	part	crypto_LUKS		5G	0	
└─myencrypteddisk	crypt	ext4		5G	0	/data4tb

As an extra step, you can validate if the data disk has any keys loaded:

```
cryptsetup luksDump /dev/VGNAME/LVNAME
```

```
cryptsetup luksDump /dev/sdd1
```

And you can check which `dm` devices are listed as `crypt`:

```
dmsetup ls --target crypt
```

## Next steps

- [Azure Disk Encryption troubleshooting](#)

# Configure LVM and RAID on encrypted devices

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article is a step-by-step process for how to perform Logical Volume Management (LVM) and RAID on encrypted devices. The process applies to the following environments:

- Linux distributions
  - RHEL 7.6+
  - Ubuntu 18.04+
  - SUSE 12+
- Azure Disk Encryption single-pass extension
- Azure Disk Encryption dual-pass extension

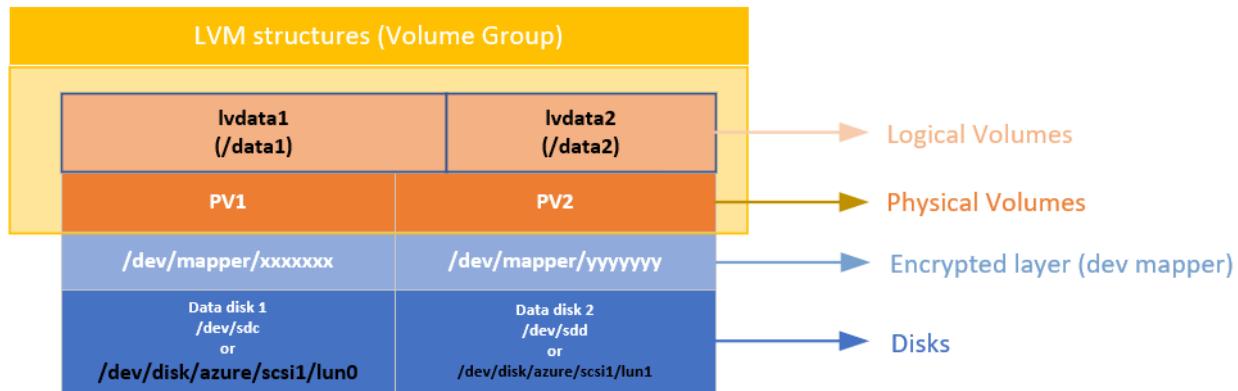
## Scenarios

The procedures in this article support the following scenarios:

- Configure LVM on top of encrypted devices (LVM-on-crypt)
- Configure RAID on top of encrypted devices (RAID-on-crypt)

After the underlying device or devices are encrypted, then you can create the LVM or RAID structures on top of that encrypted layer.

The physical volumes (PVs) are created on top of the encrypted layer. The physical volumes are used to create the volume group. You create the volumes and add the required entries on /etc/fstab.



In a similar way, the RAID device is created on top of the encrypted layer on the disks. A file system is created on top of the RAID device and added to /etc/fstab as a regular device.

## Considerations

We recommend that you use LVM-on-crypt. RAID is an option when LVM can't be used because of specific application or environment limitations.

You'll use the **EncryptFormatAll** option. For more information about this option, see [Use the EncryptFormatAll feature for data disks on Linux VMs](#).

Although you can use this method when you're also encrypting the OS, we're just encrypting data drives here.

The procedures assume that you already reviewed the prerequisites in [Azure Disk Encryption scenarios on Linux VMs](#) and in [Quickstart: Create and encrypt a Linux VM with the Azure CLI](#).

The Azure Disk Encryption dual-pass version is on a deprecation path and should no longer be used on new encryptions.

## General steps

When you're using the "on-crypt" configurations, use the process outlined in the following procedures.

### NOTE

We're using variables throughout the article. Replace the values accordingly.

### Deploy a VM

The following commands are optional, but we recommend that you apply them on a newly deployed virtual machine (VM).

PowerShell:

```
New-AzVm -ResourceGroupName ${RGNAME} `  
-Name ${VMNAME} `  
-Location ${LOCATION} `  
-Size ${VMSIZE} `  
-Image ${OSIMAGE} `  
-Credential ${creds} `  
-Verbose
```

Azure CLI:

```
az vm create \  
-n ${VMNAME} \  
-g ${RGNAME} \  
--image ${OSIMAGE} \  
--admin-username ${username} \  
--admin-password ${password} \  
-l ${LOCATION} \  
--size ${VMSIZE} \  
-o table
```

### Attach disks to the VM

Repeat the following commands for `$N` number of new disks that you want to attach to the VM.

PowerShell:

```
$storageType = 'Standard_LRS'  
$dataDiskName = ${VMNAME} + '_datadisk0'  
$diskConfig = New-AzDiskConfig -SkuName $storageType -Location $LOCATION -CreateOption Empty -DiskSizeGB 5  
$dataDisk1 = New-AzDisk -DiskName $dataDiskName -Disk $diskConfig -ResourceGroupName ${RGNAME}  
$vm = Get-AzVM -Name ${VMNAME} -ResourceGroupName ${RGNAME}  
$vm = Add-AzVMDataDisk -VM $vm -Name $dataDiskName -CreateOption Attach -ManagedDiskId $dataDisk1.Id -Lun 0  
Update-AzVM -VM ${VM} -ResourceGroupName ${RGNAME}
```

Azure CLI:

```

az vm disk attach \
-g ${RGNAME} \
--vm-name ${VMNAME} \
--name ${VMNAME}datadisk1 \
--size-gb 5 \
--new \
-o table

```

## Verify that the disks are attached to the VM

PowerShell:

```

$VM = Get-AzVM -ResourceGroupName ${RGNAME} -Name ${VMNAME}
$VM.StorageProfile.DataDisks | Select-Object Lun,Name,DiskSizeGB

```

Lun	Name	DiskSizeGB
0	vmlvmoncrypt_datadisk0	5
1	vmlvmoncrypt_datadisk1	5
2	vmlvmoncrypt_datadisk2	5
3	vmlvmoncrypt_datadisk3	5

Azure CLI:

```

az vm show -g ${RGNAME} -n ${VMNAME} --query storageProfile.dataDisks -o table

```

Lun	Name	Caching	WriteAcceleratorEnabled	CreateOption	DiskSizeGb	ToBeDetached
0	vmlvmoncrypt_datadisk0	None	False	Attach	5	False
1	vmlvmoncrypt_datadisk1	None	False	Attach	5	False
2	vmlvmoncrypt_datadisk2	None	False	Attach	5	False
3	vmlvmoncrypt_datadisk3	None	False	Attach	5	False

Portal:

Data disks

LUN	Name	Size	Storage account ...	Encryption	Host caching
0	vmlvmoncrypt_datadisk0	5 GiB	Standard HDD	Not enabled	None
1	vmlvmoncrypt_datadisk1	5 GiB	Standard HDD	Not enabled	None
2	vmlvmoncrypt_datadisk2	5 GiB	Standard HDD	Not enabled	None
3	vmlvmoncrypt_datadisk3	5 GiB	Standard HDD	Not enabled	None

OS:

```

lsblk

```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot
└─sda2	8:2	0	31.5G	0	part	/
sdb	8:16	0	32G	0	disk	
└─sdb1	8:17	0	32G	0	part	/mnt/resource
sdc	8:32	0	5G	0	disk	
sdd	8:48	0	5G	0	disk	
sde	8:64	0	5G	0	disk	
sdf	8:80	0	5G	0	disk	
sr0	11:0	1	628K	0	rom	

## Configure the disks to be encrypted

This configuration is done at the operating system level. The corresponding disks are configured for a traditional encryption through Azure Disk Encryption:

- File systems are created on top of the disks.
- Temporary mount points are created to mount the file systems.
- File systems are configured on /etc/fstab to be mounted at boot time.

Check the device letter assigned to the new disks. In this example, we're using four data disks.

```
lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot
└─sda2	8:2	0	31.5G	0	part	/
sdb	8:16	0	32G	0	disk	
└─sdb1	8:17	0	32G	0	part	/mnt/resource
sdc	8:32	0	5G	0	disk	
sdd	8:48	0	5G	0	disk	
sde	8:64	0	5G	0	disk	
sdf	8:80	0	5G	0	disk	
sr0	11:0	1	628K	0	rom	

## Create a file system on top of each disk

This command iterates the creation of an ext4 file system on each disk defined on the "in" part of the "for" cycle.

```
for disk in c d e f; do echo mkfs.ext4 -F /dev/sd${disk}; done |bash
```

```
[root@vmlvmoncrypt ~]# for disk in c d e; do echo mkfs.ext4 -F /dev/sd${disk}; done |bash
mke2fs 1.42.9 (28-Dec-2013)
Discarding device blocks: done
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
327680 inodes, 1310720 blocks
65536 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1342177280
40 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
      32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Find the universally unique identifier (UUID) of the file systems that you recently created, create a temporary folder, add the corresponding entries on /etc/fstab, and mount all the file systems.

This command also iterates on each disk defined on the "in" part of the "for" cycle:

```

for disk in c d e f; do diskuuid=$(blkid -s UUID -o value /dev/sd${disk}); \
mkdir /tempdata${disk}; \
echo "UUID=${diskuuid} /tempdata${disk} ext4 defaults,nofail 0 0" >> /etc/fstab; \
mount -a; \
done

```

### Verify that the disks are mounted properly

```
lsblk
```

```
[root@vmlvmoncrypt ~]# lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0      2:0    1   4K  0 disk 
sda      8:0    0  32G  0 disk 
└─sda1   8:1    0 500M  0 part /boot
└─sda2   8:2    0 31.5G  0 part /
sdb      8:16   0  32G  0 disk 
└─sdb1   8:17   0  32G  0 part /mnt/resource
sdc      8:32   0   5G  0 disk /tempdatac
sdd      8:48   0   5G  0 disk /tempdatad
sde      8:64   0   5G  0 disk /tempdatae
 sdf      8:80   0   5G  0 disk /tempdataf
sr0     11:0   1  628K  0 rom
```

Also verify that the disks are configured:

```
cat /etc/fstab
```

```
[root@vmlvmoncrypt ~]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Thu Jun 20 18:47:49 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=52e42afc-2da4-4391-bbc0-92f2a1b45c62 /          xfs      defaults        0 0
UUID=a8fbef35-ff5e-4515-bb7e-c408c118d4c0 /boot        xfs      defaults        0 0
UUID=e347f00e-02ca-4b9b-9aee-1295c09ed634 /tempdatac ext4 defaults,nofail 0 0
UUID=1ee5221f-201a-4da7-a0df-fb0c906ac323 /tempdatad ext4 defaults,nofail 0 0
UUID=a16ab0a2-0e47-4a04-a5a9-28618e6377ba /tempdatae ext4 defaults,nofail 0 0
UUID=5a98ba0d-9b92-44c8-ac7d-9335e56ab0f3 /tempdataf ext4 defaults,nofail 0 0
```

### Encrypt the data disks

PowerShell using a key encryption key (KEK):

```
$sequenceVersion = [Guid]::NewGuid()
Set-AzVMDiskEncryptionExtension -ResourceGroupName $RGNAME ` 
-VMName ${VMNAME} ` 
-DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl ` 
-DiskEncryptionKeyId $KeyVaultResourceId ` 
-KeyEncryptionKeyUrl $keyEncryptionKeyUrl ` 
-KeyEncryptionKeyVaultId $KeyVaultResourceId ` 
-VolumeType 'DATA' ` 
-EncryptFormatAll ` 
-SequenceVersion $sequenceVersion ` 
-skipVmBackup;
```

Azure CLI using a KEK:

```
az vm encryption enable \
--resource-group ${RGNAME} \
--name ${VMNAME} \
--disk-encryption-keyvault ${KEYVAULTNAME} \
--key-encryption-key ${KEYNAME} \
--key-encryption-keyvault ${KEYVAULTNAME} \
--volume-type "DATA" \
--encrypt-format-all \
-o table
```

## Verify the encryption status

Continue to the next step only when all the disks are encrypted.

PowerShell:

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName ${RGNAME} -VMName ${VMNAME}
```

```
OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage        : Encryption succeeded for data volumes
```

Azure CLI:

```
az vm encryption show -n ${VMNAME} -g ${RGNAME} -o table
```

```
user@Azure>az vm encryption show -n ${VMNAME} -g ${RGNAME} -o table
Status           Message
-----
Provisioning succeeded Encryption succeeded for data volumes
```

Portal:

Data disks						
LUN	Name	Size	Storage account ...	Encryption	Host caching	
0	vmlvmoncrypt_datadisk0	5 GiB	Standard HDD	<b>Enabled</b>	None	
1	vmlvmoncrypt_datadisk1	5 GiB	Standard HDD	<b>Enabled</b>	None	
2	vmlvmoncrypt_datadisk2	5 GiB	Standard HDD	<b>Enabled</b>	None	
3	vmlvmoncrypt_datadisk3	5 GiB	Standard HDD	<b>Enabled</b>	None	

OS level:

```
lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└ sda1	8:1	0	500M	0	part	/boot
└ sda2	8:2	0	31.5G	0	part	/
sdb	8:16	0	32G	0	disk	
└ sdb1	8:17	0	32G	0	part	
└ resourceencrypt	253:4	0	32G	0	crypt	/mnt/resource
sdc	8:32	0	48M	0	disk	
└ sdc1	8:33	0	47M	0	part	/mnt/azure_bek_disk
sdd	8:48	0	5G	0	disk	
└ c49ff535-1df9-45ad-9dad-f0846509f052	253:0	0	5G	0	crypt	/tempdatac
sde	8:64	0	5G	0	disk	
└ 6712ad6f-65ce-487b-aa52-462f381611a1	253:1	0	5G	0	crypt	/tempdatad
sdf	8:80	0	5G	0	disk	
└ ea607dfd-c396-48d6-bc54-603cf741bc2a	253:2	0	5G	0	crypt	/tempdatae
sdg	8:96	0	5G	0	disk	
└ 4159c60a-a546-455b-985f-92865d51158c	253:3	0	5G	0	crypt	/tempdataf
sr0	11:0	1	1024M	0	rom	

The extension will add the file systems to /var/lib/azure\_disk\_encryption\_config/azure\_crypt\_mount (an old encryption) or to /etc/crypttab (new encryptions).

#### NOTE

Do not modify any of these files.

This file will take care of activating these disks during the boot process so that LVM or RAID can use them later.

Don't worry about the mount points on this file. Azure Disk Encryption will lose the ability to get the disks mounted as a normal file system after we create a physical volume or a RAID device on top of those encrypted devices. (This will remove the file system format that we used during the preparation process.)

#### Remove the temporary folders and temporary fstab entries

You unmount the file systems on the disks that will be used as part of LVM.

```
for disk in c d e f; do umount /tempdata${disk}; done
```

And remove the /etc/fstab entries:

```
vi /etc/fstab
```

#### Verify that the disks are not mounted and that the entries on /etc/fstab were removed

```
lsblk
```

```
[root@vmlvmoncrypt ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0            2:0    1   4K  0 disk 
sda            8:0    0  32G  0 disk 
└─sda1         8:1    0 500M  0 part /boot
└─sda2         8:2    0 31.5G 0 part /
sdb            8:16   0  32G  0 disk 
└─sdb1         8:17   0  32G  0 part 
  └─resourceencrypt 253:4  0  32G  0 crypt /mnt/resource
sdc            8:32   0  48M  0 disk 
└─sdc1         8:33   0  47M  0 part /mnt/azure_bek_disk
sdd            8:48   0   5G  0 disk 
└─c49ff535-1df9-45ad-9dad-f0846509f052 253:0  0   5G  0 crypt
sde            8:64   0   5G  0 disk 
└─6712ad6f-65ce-487b-aa52-462f381611a1 253:1  0   5G  0 crypt
sdf            8:80   0   5G  0 disk 
└─ea607dfd-c396-48d6-bc54-603cf741bc2a 253:2  0   5G  0 crypt
sdg            8:96   0   5G  0 disk 
└─4159c60a-a546-455b-985f-92865d51158c 253:3  0   5G  0 crypt
sr0           11:0   1 1024M 0 rom
```

And verify that the disks are configured:

```
cat /etc/fstab
```

```
[root@vmlvmoncrypt ~]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Thu Jun 20 18:47:49 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=52e42afc-2da4-4391-bbc0-92f2a1b45c62 /          xfs      defaults        0 0
UUID=a8fbef35-ff5e-4515-bb7e-c408c118d4c0 /boot       xfs      defaults        0 0
LABEL=BEK\040VOLUME /mnt/azure_bek_disk auto defaults,discard,nofail 0 0
/dev/mapper/resourcecrypt /mnt/resource auto defaults,discard,nofail 0 0
```

## Steps for LVM-on-crypt

Now that the underlying disks are encrypted, you can create the LVM structures.

Instead of using the device name, use the /dev/mapper paths for each of the disks to create a physical volume (on the crypt layer on top of the disk, not on the disk itself).

### Configure LVM on top of the encrypted layers

#### Create the physical volumes

You'll get a warning that asks if it's OK to wipe out the file system signature. Continue by entering **y**, or use **echo "y"** as shown:

```
echo "y" | pvcreate /dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052
echo "y" | pvcreate /dev/mapper/6712ad6f-65ce-487b-aa52-462f381611a1
echo "y" | pvcreate /dev/mapper/ea607dfd-c396-48d6-bc54-603cf741bc2a
echo "y" | pvcreate /dev/mapper/4159c60a-a546-455b-985f-92865d51158c
```

```
[root@vmlvmoncrypt ~]# echo "y" | pvcreate /dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052
WARNING: ext4 signature detected on /dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052 at offset 1080. Wipe it? [y/n]: t
Physical volume "/dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052" successfully created.
```

#### NOTE

The /dev/mapper/device names here need to be replaced for your actual values based on the output of **lsblk**.

### Verify the information for physical volumes

```
pvs
```

```
[root@vmlvmoncrypt ~]# pvs
PV                                         VG Fmt Attr PSize PFree
/dev/mapper/4159c60a-a546-455b-985f-92865d51158c  lvm2 --- <5.00g <5.00g
/dev/mapper/6712ad6f-65ce-487b-aa52-462f381611a1  lvm2 --- <5.00g <5.00g
/dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052  lvm2 --- <5.00g <5.00g
/dev/mapper/ea607dfd-c396-48d6-bc54-603cf741bc2a  lvm2 --- <5.00g <5.00g
```

#### Create the volume group

Create the volume group by using the same devices already initialized:

```
vgcreate vgdata /dev/mapper/
```

#### Check the information for the volume group

```
vgdisplay -v vgdata
```

```
pvs
```

```
[root@vmlvmoncrypt ~]# pvs
PV                                         VG      Fmt  Attr PSize PFree
/dev/mapper/4159c60a-a546-455b-985f-92865d51158c vgdata  lvm2 a-- <5.00g <5.00g
/dev/mapper/6712ad6f-65ce-487b-aa52-462f381611a1 vgdata  lvm2 a-- <5.00g <5.00g
/dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052 vgdata  lvm2 a-- <5.00g <5.00g
/dev/mapper/ea607dfd-c396-48d6-bc54-603cf741bc2a vgdata  lvm2 a-- <5.00g <5.00g
```

#### Create logical volumes

```
lvcreate -L 10G -n lvdata1 vgdata
lvcreate -L 7G -n lvdata2 vgdata
```

#### Check the created logical volumes

```
lvdisplay
lvdisplay vgdata/lvdata1
lvdisplay vgdata/lvdata2
```

```
[root@vmlvmoncrypt ~]# lvdisplay
--- Logical volume ---
LV Path /dev/vgdata/lvdata1
LV Name lvdata1
VG Name vgdata
LV UUID 3tyHXl-txNi-zo22-weaP-q7N3-zfy-e-ggJ18D
LV Write Access read/write
LV Creation host, time vmlvmoncrypt, 2020-03-18 01:45:26 +0000
LV Status available
# open 0
LV Size 10.00 GiB
Current LE 2560
Segments 3
Allocation inherit
Read ahead sectors auto
- currently set to 8192
Block device 253:5

--- Logical volume ---
LV Path /dev/vgdata/lvdata2
LV Name lvdata2
VG Name vgdata
LV UUID o2SEEz-dDSL-YxmC-a441-LiV7-b7J0-dwyFso
LV Write Access read/write
LV Creation host, time vmlvmoncrypt, 2020-03-18 01:45:38 +0000
LV Status available
# open 0
LV Size 7.00 GiB
Current LE 1792
Segments 2
Allocation inherit
Read ahead sectors auto
- currently set to 8192
Block device 253:6
```

#### Create file systems on top of the structures for logical volumes

```
echo "yes" | mkfs.ext4 /dev/vgdata/lvdata1
echo "yes" | mkfs.ext4 /dev/vgdata/lvdata2
```

#### Create the mount points for the new file systems

```
mkdir /data0
mkdir /data1
```

#### Add the new file systems to /etc/fstab and mount them

```
echo "/dev/mapper/vgdata-lvdata1 /data0 ext4 defaults,nofail 0 0" >>/etc/fstab
echo "/dev/mapper/vgdata-lvdata2 /data1 ext4 defaults,nofail 0 0" >>/etc/fstab
mount -a
```

#### Verify that the new file systems are mounted

```
lsblk -fs
df -h
```

```

vgdata-lvdata1          ext4
└─c49ff535-1df9-45ad-9dad-f0846509f052 LVM2_member
  └─sdd
    └─6712ad6f-65ce-487b-aa52-462f381611a1 LVM2_member
      └─sde
        └─4159c60a-a546-455b-985f-92865d51158c LVM2_member
          └─sdg
            └─crypto_LUKS
vgdata-lvdata2          ext4
└─c49ff535-1df9-45ad-9dad-f0846509f052 LVM2_member
  └─sdd
    └─ea607dfd-c396-48d6-bc54-603cf741bc2a LVM2_member
      └─sdf
        └─crypto_LUKS

```

5b3ca005-5eee-4f6a-8177-9961e47d8a89 /data0  
RwpOHw-mPar-iKOi-ATGR-xsgd-xXO8-TBvf1v  
dfbb7ff0-c973-4b3e-aa76-4b6888ce8667  
BLktAV-kz6P-UDgL-0UJ0-3G3J-97Dk-8AiA8b  
f99743c8-957d-4a9b-b6b5-2ac64e765d3f  
hTQBWm-vidY-ZAH2-1bXz-S12H-MnCE-92tbfX  
8b9089da-ccb3-46a0-a7e3-082559e79659  
2b94aa7d-bdb2-4fc7-bff1-032bf8227647 /data1  
RwpOHw-mPar-iKOi-ATGR-xsgd-xXO8-TBvf1v  
dfbb7ff0-c973-4b3e-aa76-4b6888ce8667  
uayh0G-BDQW-bzup-puQp-eyE8-D3Xp-S6We04  
8d5faa52-282f-4609-9bc3-f3077c910982

On this variation of **lsblk**, we're listing the devices showing the dependencies in reverse order. This option helps to identify the devices grouped by the logical volume instead of the original /dev/sd[disk] device names.

It's important to make sure that the **nofail** option is added to the mount point options of the LVM volumes created on top of a device encrypted through Azure Disk Encryption. It prevents the OS from getting stuck during the boot process (or in maintenance mode).

If you don't use the **nofail** option:

- The OS will never get into the stage where Azure Disk Encryption is started and the data disks are unlocked and mounted.
- The encrypted disks will be unlocked at the end of the boot process. The LVM volumes and file systems will be automatically mounted until Azure Disk Encryption unlocks them.

You can test rebooting the VM and validate that the file systems are also automatically getting mounted after boot time. This process might take several minutes, depending on the number and sizes of file systems.

#### **Reboot the VM and verify after reboot**

```
shutdown -r now
```

```
lsblk
df -h
```

## Steps for RAID-on-crypt

Now that the underlying disks are encrypted, you can continue to create the RAID structures. The process is the same as the one for LVM, but instead of using the device name, use the /dev/mapper paths for each disk.

#### **Configure RAID on top of the encrypted layer of the disks**

```
mdadm --create /dev/md10 \
--level 0 \
--raid-devices=4 \
/dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052 \
/dev/mapper/6712ad6f-65ce-487b-aa52-462f381611a1 \
/dev/mapper/ea607dfd-c396-48d6-bc54-603cf741bc2a \
/dev/mapper/4159c60a-a546-455b-985f-92865d51158c
```

```
[root@vmlvmoncrypt ~]# mdadm --create /dev/md10 \
> --level 0 \
> --raid-devices=4 \
> /dev/mapper/c49ff535-1df9-45ad-9dad-f0846509f052 \
> /dev/mapper/6712ad6f-65ce-487b-aa52-462f381611a1 \
> /dev/mapper/ea607dfd-c396-48d6-bc54-603cf741bc2a \
> /dev/mapper/4159c60a-a546-455b-985f-92865d51158c
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md10 started.
```

#### NOTE

The /dev/mapper/device names here need to be replaced with your actual values, based on the output of `lsblk`.

#### Check/monitor RAID creation

```
watch -n1 cat /proc/mdstat  
mdadm --examine /dev/mapper/[]  
mdadm --detail /dev/md10
```

```
[root@vmlvmoncrypt ~]# mdadm --detail /dev/md10  
/dev/md10:  
    Version : 1.2  
Creation Time : Wed Mar 18 02:15:42 2020  
    Raid Level : raid0  
    Array Size : 20942848 (19.97 GiB 21.45 GB)  
    Raid Devices : 4  
    Total Devices : 4  
        Persistence : Superblock is persistent  
  
        Update Time : Wed Mar 18 02:15:42 2020  
            State : clean  
    Active Devices : 4  
Working Devices : 4  
Failed Devices : 0  
Spare Devices : 0  
  
    Chunk Size : 512K  
  
Consistency Policy : none  
  
                Name : vmlvmoncrypt:10 (local to host vmlvmoncrypt)  
                UUID : 62fbf2ff:0dae41b1:96c61bd5:b3119c49  
                Events : 0  
  
      Number  Major  Minor  RaidDevice State  
        0      253       0        0  active sync  /dev/dm-0  
        1      253       1        1  active sync  /dev/dm-1  
        2      253       2        2  active sync  /dev/dm-2  
        3      253       4        3  active sync  /dev/dm-4
```

#### Create a file system on top of the new RAID device

```
mkfs.ext4 /dev/md10
```

Create a new mount point for the file system, add the new file system to /etc/fstab, and mount it:

#### NOTE

This cycle iterates only on one device for this particular example, is built this way to be used for multiple md devices if needed.

```
for device in md10; do diskuuid=$(blkid -s UUID -o value /dev/${device}); \  
mkdir /raiddata; \  
echo "UUID=${diskuuid} /raiddata ext4 defaults,nofail 0 0" >> /etc/fstab; \  
mount -a; \  
done
```

Verify that the new file system is mounted:

```
lsblk -fs  
df -h
```

```
md10  
└─c49ff535-1df9-45ad-9dad-f0846509f052 ext4  
    └─sdd  
        └─sde  
            └─ea607dfd-c396-48d6-bc54-603cf741bc2a  
                └─sdf  
                    └─4159c60a-a546-455b-985f-92865d51158c  
                        └─sda  
                            └─crypto_LUKS  
                                └─vmlvmoncrypt:10  
                                    └─c72438c2-8496-4d5b-aa55-90bf979b080e /raiddata
```

It's important to make sure that the **nofail** option is added to the mount point options of the RAID volumes created on top of a device encrypted through Azure Disk Encryption. It prevents the OS from getting stuck during the boot process (or in maintenance mode).

If you don't use the **nofail** option:

- The OS will never get into the stage where Azure Disk Encryption is started and the data disks are unlocked and mounted.
- The encrypted disks will be unlocked at the end of the boot process. The RAID volumes and file systems will be automatically mounted until Azure Disk Encryption unlocks them.

You can test rebooting the VM and validate that the file systems are also automatically getting mounted after boot time. This process might take several minutes, depending on the number and sizes of file systems.

```
shutdown -r now
```

And when you can log in:

```
lsblk  
df -h
```

## Next steps

- [Resize logical volume management devices encrypted with Azure Disk Encryption](#)
- [Azure Disk Encryption troubleshooting](#)

# How to resize logical volume management devices that use Azure Disk Encryption

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In this article, you'll learn how to resize data disks that use Azure Disk Encryption. To resize the disks, you'll use logical volume management (LVM) on Linux. The steps apply to multiple scenarios.

You can use this resizing process in the following environments:

- Linux distributions:
  - Red Hat Enterprise Linux (RHEL) 7 or later
  - Ubuntu 16 or later
  - SUSE 12 or later
- Azure Disk Encryption versions:
  - Single-pass extension
  - Dual-pass extension

## Prerequisites

This article assumes that you have:

- An existing LVM configuration. For more information, see [Configure LVM on a Linux VM](#).
- Disks that are already encrypted by Azure Disk Encryption. For more information, see [Configure LVM and RAID on encrypted devices](#).
- Experience using Linux and LVM.
- Experience using `/dev/disk/scsi1/paths` for data disks on Azure. For more information, see [Troubleshoot Linux VM device name problems](#).

## Scenarios

The procedures in this article apply to the following scenarios:

- Traditional LVM and LVM-on-crypt configurations
- Traditional LVM encryption
- LVM-on-crypt

### Traditional LVM and LVM-on-crypt configurations

Traditional LVM and LVM-on-crypt configurations extend a logical volume (LV) when the volume group (VG) has available space.

### Traditional LVM encryption

In traditional LVM encryption, LVs are encrypted. The whole disk isn't encrypted.

By using traditional LVM encryption, you can:

- Extend the LV when you add a new physical volume (PV).
- Extend the LV when you resize an existing PV.

## LVM-on-crypt

The recommended method for disk encryption is LVM-on-encrypt. This method encrypts the entire disk, not just the LV.

By using LVM-on-crypt, you can:

- Extend the LV when you add a new PV.
- Extend the LV when you resize an existing PV.

### NOTE

We don't recommend mixing traditional LVM encryption and LVM-on-crypt on the same VM.

The following sections provide examples of how to use LVM and LVM-on-crypt. The examples use preexisting values for disks, PVs, VGs, LVs, file systems, universally unique identifiers (UUIDs), and mount points. Replace these values with your own values to fit your environment.

### Extend an LV when the VG has available space

The traditional way to resize LVs is to extend an LV when the VG has space available. You can use this method for nonencrypted disks, traditional LVM-encrypted volumes, and LVM-on-crypt configurations.

1. Verify the current size of the file system that you want to increase:

```
df -h /mountpoint
```

```
[root@azurervm]# df -h /datalv02
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/78b259a1-956b-4f3a-8e6b-c05bf5040bc5  974M  958M     0 100% /datalv02
```

2. Verify that the VG has enough space to increase the LV:

```
vgs
```

```
[root@azurervm]# vgs
VG #PV #LV #SN Attr   VSize   VFree
datavg    3    4    0 wz--n- <14.99g <4.99g
```

You can also use `vgdisplay`:

```
vgdisplay vgname
```

```
[root@azurervm]# vgdisplay datavg
--- Volume group ---
VG Name           datavg
System ID
Format          lvm2
Metadata Areas    3
Metadata Sequence No 5
VG Access        read/write
VG Status         resizable
MAX LV            0
Cur LV            4
Open LV           4
Max PV            0
Cur PV            3
Act PV            3
VG Size          <14.99 GiB
PE Size          4.00 MiB
Total PE          3837
Alloc PE / Size  2560 / 10.00 GiB
Free PE / Size   1277 / <4.99 GiB
VG UUID          jvc1Kz-FXLd-dNjC-14q7-k85I-Pp8z-bMqFLY
```

3. Identify which LV needs to be resized:

```
lsblk
```

```
[root@azurervm]# lsblk
NAME                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
fd0                  2:0    1   4K  0 disk
sda                  8:0    0 32G  0 disk
└─sda1                8:1    0 500M 0 part  /boot
└─sda2                8:2    0 31.5G 0 part  /
sdb                  8:16   0 32G  0 disk
└─sdb1                8:17   0 32G  0 part  /mnt/resource
sdc                  8:32   0 48M  0 disk
└─sdc1                8:33   0 46M  0 part  /mnt/azure_bek_disk
sdd                  8:48   0 5G   0 disk
└─datavg-datalv01      253:0  0 1G   0 lvm
  └─55b4af2-a160-426b-9bd7-588af6c46e9b 253:4  0 1022M 0 crypt  /datalv01
└─datavg-datalv02      253:1  0 1G   0 lvm
  └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5 253:5  0 1022M 0 crypt  /datalv02
└─datavg-datalv03      253:2  0 1G   0 lvm
  └─70abfc58-b0fd-441b-8b77-f86c9249af5e 253:6  0 1022M 0 crypt  /datalv03
sde                  8:64   0 5G   0 disk
└─datavg-datalv04      253:3  0 7G   0 lvm
  └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7  0 7G   0 crypt  /datalv04
 sdf                  8:80   0 5G   0 disk
└─datavg-datalv04      253:3  0 7G   0 lvm
  └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7  0 7G   0 crypt  /datalv04
sr0                 11:0   1 1024M 0 rom
```

For LVM-on-crypt, the difference is that this output shows that the encrypted layer is at the disk level.

```
[root@vmresizelv ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0            2:0    1   4K  0 disk 
sda            8:0    0   32G  0 disk 
└─sda1         8:1    0   500M 0 part /boot
      └─sda2         8:2    0   31.5G 0 part /
sdb            8:16   0   32G  0 disk 
└─sdb1         8:17   0   32G  0 part 
  └─resourcecrypt 253:0   0   32G  0 crypt /mnt/resource
sdc            8:32   0   48M  0 disk 
└─sdc1         8:33   0   47M  0 part /mnt/azure_bek_disk
sde            8:64   0     5G  0 disk 
└─e19fcd77-e974-4e5c-a874-e78e4b6d2f48 253:1   0   5G  0 crypt
  ├─datavg-datalv01 253:4   0   1G  0 lvm  /datalv01
  └─datavg-datalv02 253:5   0   6G  0 lvm  /datalv02
    ├─datavg-datalv03 253:6   0   4G  0 lvm  /datalv03
    └─datavg-datalv04 253:7   0   3G  0 lvm  /datalv04
sdf            8:80   0   5G  0 disk 
└─49de5df0-1c65-48bf-809c-588805eda921 253:2   0   5G  0 crypt
  ├─datavg-datalv02 253:5   0   6G  0 lvm  /datalv02
  └─datavg-datalv04 253:7   0   3G  0 lvm  /datalv04
sdg            8:96   0   5G  0 disk 
└─8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 253:3   0   5G  0 crypt
  ├─datavg-datalv03 253:6   0   4G  0 lvm  /datalv03
  └─datavg-datalv04 253:7   0   3G  0 lvm  /datalv04
```

4. Check the LV size:

```
lvdisplay lvname
```

```
[root@azurervm]# lvdisplay /dev/datavg/datalv02
--- Logical volume ---
LV Path           /dev/datavg/datalv02
LV Name          datalv02
VG Name          datavg
LV UUID          Isacy4-ukYq-UKCB-tG48-tjfc-LWlr-kWTyfy
LV Write Access  read/write
LV Creation host, time  vmresizelv, 2020-09-18 20:13:14 +0000
LV Status        available
# open           1
LV Size          1.00 GiB
Current LE       256
Segments         1
Allocation       inherit
Read ahead sectors auto
- currently set to 8192
Block device     253:1
```

5. Increase the LV size by using `-r` to resize the file system online:

```
lvextend -r -L +2G /dev/vgname/lvname
```

```
[root@azurervm]# lvextend -r -L +2G /dev/datavg/datalv02
  size of logical volume datavg/datalv02 changed from 1.00 GiB (256 extents) to 3.00 GiB (768 extents).
Logical volume datavg/datalv02 successfully resized.
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/78b259a1-956b-4f3a-8e6b-c05bf5040bc5 is mounted on /datalv02; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/mapper/78b259a1-956b-4f3a-8e6b-c05bf5040bc5 is now 785920 blocks long.
```

6. Verify the new sizes for the LV and the file system:

```
df -h /mountpoint
```

```
[root@azurervm]# df -h /datalv02/
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/78b259a1-956b-4f3a-8e6b-c05bf5040bc5  3.0G  959M  1.9G  34% /datalv02
```

The size output indicates that the LV and file system were successfully resized.

You can check the LV information again to confirm the changes at the level of the LV:

```
lvdisplay lvname
```

```
[root@azurevm]# lvdisplay /dev/datavg/datalv02
--- Logical volume ---
LV Path              /dev/datavg/datalv02
LV Name             datalv02
VG Name              datavg
LV UUID             Isacy4-ukYq-UKCB-tG48-tjfc-LWlr-kWTyfy
LV Write Access     read/write
LV Creation host, time vmresizelv, 2020-09-18 20:13:14 +0000
LV Status            available
# open                1
LV Size              3.00 GiB
Current LE           768
Segments              3
Allocation            inherit
Read ahead sectors   auto
- currently set to    8192
Block device          253:1
```

#### Extend a traditional LVM volume by adding a new PV

When you need to add a new disk to increase the VG size, extend your traditional LVM volume by adding a new PV.

1. Verify the current size of the file system that you want to increase:

```
df -h /mountpoint
```

```
[root@azurevm]# df -h /datalv01
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/55b4af2-a160-426b-9bd7-588af6c46e9b  974M  958M      0 100% /datalv01
```

2. Verify the current PV configuration:

```
pvs
```

```
[root@azurevm]# pvs
PV      VG      Fmt  Attr PSize  PFree
/dev/sdd  datavg lvm2 a-- <5.00g    0
/dev/sde  datavg lvm2 a-- <5.00g    0
/dev/sdf  datavg lvm2 a-- <5.00g    0
```

3. Check the current VG information:

```
vgs
```

```
[root@azurevm]# vgs
VG      #PV  #LV  #SN Attr   VSize  VFree
datavg    3    4    0 wz--n- <14.99g    0
```

4. Check the current disk list. Identify data disks by checking the devices in `/dev/disk/azure/scsi1/`.

```
ls -l /dev/disk/azure/scsi1/
```

```
[root@azurervm]# ls -l /dev/disk/azure/scsi1/
total 0
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun0 -> ../../.../sdd
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun1 -> ../../.../sde
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun2 -> ../../.../sdf
```

5. Check the output of `lsblk`:

```
lsblk
```

```
[root@azurervm]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
fd0           2:0    1   4K  0 disk
sda           8:0    0  32G  0 disk
└─sda1        8:1    0 500M  0 part  /boot
   └─sda2        8:2    0 31.5G 0 part  /
sdb           8:16   0  32G  0 disk
└─sdb1        8:17   0 32G  0 part  /mnt/resource
sdc           8:32   0  48M  0 disk
└─sdc1        8:33   0  46M  0 part  /mnt/azure_bek_disk
sdd           8:48   0   5G  0 disk
  ├─datavg-datalv01  253:0   0  1G  0 lvm
  │  ├─55b4af2d-a160-426b-9bd7-588af6c46e9b  253:4   0 1022M 0 crypt /datalv01
  │  ├─datavg-datalv02  253:1   0  6G  0 lvm
  │  └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5  253:5   0  6G  0 crypt /datalv02
  ├─datavg-datalv03  253:2   0  1G  0 lvm
  └─70abfc58-b0fd-441b-8b77-f86c9249af5e  253:6   0 1022M 0 crypt /datalv03
sde           8:64   0   5G  0 disk
  └─datavg-datalv04  253:3   0  7G  0 lvm
    └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c  253:7   0  7G  0 crypt /datalv04
sdf           8:80   0   5G  0 disk
  └─datavg-datalv02  253:1   0  6G  0 lvm
    └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5  253:5   0  6G  0 crypt /datalv02
  └─datavg-datalv04  253:3   0  7G  0 lvm
    └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c  253:7   0  7G  0 crypt /datalv04
sr0          11:0   1 1024M 0 rom
```

6. Attach the new disk to the VM by following the instructions in [Attach a data disk to a Linux VM](#).

7. Check the disk list, and notice the new disk.

```
ls -l /dev/disk/azure/scsi1/
```

```
[root@azurervm]# ls -l /dev/disk/azure/scsi1/
total 0
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun0 -> ../../.../sdd
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun1 -> ../../.../sde
lrwxrwxrwx. 1 root root 12 Sep 18 21:43 lun2 -> ../../.../sdf
lrwxrwxrwx. 1 root root 12 Sep 18 22:20 lun3 -> ../../.../sdg
```

```
lsblk
```

```
[root@azurervm]# lsblk
NAME          MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
fd0            2:0    1    4K  0 disk 
sda            8:0    0   32G  0 disk 
└─sda1         8:1    0   500M 0 part /boot
└─sda2         8:2    0   31.5G 0 part /
sdb            8:16   0   32G  0 disk 
└─sdb1         8:17   0   32G  0 part /mnt/resource
sdc            8:32   0   48M  0 disk 
└─sdc1         8:33   0   46M  0 part /mnt/azure_bek_disk
sdd            8:48   0    5G  0 disk 
└─datavg-datalv01 253:0   0   1G  0 lvm 
  └─55b4af2-a160-426b-9bd7-588af6c46e9b 253:4   0 1022M 0 crypt /datalv01
└─datavg-datalv02 253:1   0   6G  0 lvm 
  └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5 253:5   0   6G  0 crypt /datalv02
└─datavg-datalv03 253:2   0   1G  0 lvm 
  └─70abfc58-b0fd-441b-8b77-f86c9249af5e 253:6   0 1022M 0 crypt /datalv03
sde            8:64   0    5G  0 disk 
└─datavg-datalv04 253:3   0   7G  0 lvm 
  └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7   0   7G  0 crypt /datalv04
sdf            8:80   0    5G  0 disk 
└─datavg-datalv02 253:1   0   6G  0 lvm 
  └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5 253:5   0   6G  0 crypt /datalv02
└─datavg-datalv04 253:3   0   7G  0 lvm 
  └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7   0   7G  0 crypt /datalv04
sdg            8:96   0    5G  0 disk 
sr0           11:0   1 1024M 0 rom
```

8. Create a new PV on top of the new data disk:

```
pvcreate /dev/newdisk
```

```
[root@azurervm]# pvcreate /dev/sdg
  Physical volume "/dev/sdg" successfully created.
[root@azurervm]#
```

This method uses the whole disk as a PV without a partition. Alternatively, you can use `fdisk` to create a partition and then use that partition for `pvcreate`.

9. Verify that the PV was added to the PV list:

```
pvs
```

```
[root@azurervm]# pvs
  PV        VG      Fmt  Attr PSize  PFree
  /dev/sdd  datavg lvm2 a-- <5.00g    0
  /dev/sde  datavg lvm2 a-- <5.00g    0
  /dev/sdf  datavg lvm2 a-- <5.00g    0
  /dev/sdg          lvm2 ---  5.00g  5.00g
```

10. Extend the VG by adding the new PV to it:

```
vgextend vgname /dev/newdisk
```

```
[root@azurervm]# vgextend datavg /dev/sdg
  Volume group "datavg" successfully extended
```

11. Check the new VG size:

```
vgs
```

```
[root@azurervm]# vgs
  VG     #PV #LV #SN Attr   VSize  VFree
  datavg    4    4    0 wz--n- 19.98g <5.00g
```

12. Use `lsblk` to identify the LV that needs to be resized:

```
lsblk
```

```
[root@azurervm]# lsblk
NAME                           MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
fd0                            2:0    1   4K  0 disk
sda                            8:0    0  32G  0 disk
└─sda1                          8:1    0 500M  0 part  /boot
    └─sda2                        8:2    0 31.5G 0 part  /
sdb                            8:16   0  32G  0 disk
└─sdb1                          8:17   0 32G  0 part  /mnt/resource
sdc                            8:32   0  48M  0 disk
└─sdc1                          8:33   0  46M  0 part  /mnt/azure_bek_disk
sdd                            8:48   0   5G  0 disk
└─datavg-datalv01              253:0   0   1G  0 lvm
    └─55b4af2-a160-426b-9bd7-588af6c46e9b 253:4   0 1022M 0 crypt /datalv01
datavg-datalv02                253:1   0   6G  0 lvm
    └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5 253:5   0   6G  0 crypt /datalv02
datavg-datalv03                253:2   0   1G  0 lvm
    └─70abfc58-b0fd-441b-8b77-f86c9249af5e 253:6   0 1022M 0 crypt /datalv03
sde                            8:64   0   5G  0 disk
└─datavg-datalv04              253:3   0   7G  0 lvm
    └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7   0   7G  0 crypt /datalv04
sdf                            8:80   0   5G  0 disk
└─datavg-datalv02              253:1   0   6G  0 lvm
    └─78b259a1-956b-4f3a-8e6b-c05bf5040bc5 253:5   0   6G  0 crypt /datalv02
datavg-datalv04                253:3   0   7G  0 lvm
    └─e5dee6bc-0f22-4579-b6c0-6103a9aba86c 253:7   0   7G  0 crypt /datalv04
sdg                            8:96   0   5G  0 disk
```

13. Extend the LV size by using `-r` to increase the file system online:

```
lvextend -r -L +2G /dev/vgname/lvname
```

```
[root@azurervm]# lvextend -r -L +2G /dev/datavg/datalv01
  Size of logical volume datavg/datalv01 changed from 1.00 GiB (256 extents) to 3.00 GiB (768 extents).
Logical volume datavg/datalv01 successfully resized.
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/55b4af2-a160-426b-9bd7-588af6c46e9b is mounted on /datalv01; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/mapper/55b4af2-a160-426b-9bd7-588af6c46e9b is now 785920 blocks long.
```

14. Verify the new sizes of the LV and file system:

```
df -h /mountpoint
```

```
[root@azurervm]# df -h /datalv01
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/55b4af2-a160-426b-9bd7-588af6c46e9b  3.0G  959M  1.9G  34% /datalv01
```

#### IMPORTANT

When Azure Data Encryption is used on traditional LVM configurations, the encrypted layer is created at the LV level, not at the disk level.

At this point, the encrypted layer is expanded to the new disk. The actual data disk has no encryption settings at the platform level, so its encryption status isn't updated.

These are some of the reasons why LVM-on-crypt is the recommended approach.

15. Check the encryption information from the portal:

LUN ⓘ	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (Mbps)	Encryption ⓘ
0	vmresizely_datadisk0	Standard HDD	5	500	60	SSE with PMK & ADE
1	vmresizely_datadisk1	Standard HDD	5	500	60	SSE with PMK & ADE
2	vmresizely_datadisk2	Standard HDD	5	500	60	SSE with PMK & ADE
3	vmresizely_datadisk3	Standard HDD	5	500	60	SSE with PMK

To update the encryption settings on the disk, add a new LV and enable the extension on the VM.

16. Add a new LV, create a file system on it, and add it to `/etc/fstab`.
17. Set the encryption extension again. This time you'll stamp the encryption settings on the new data disk at the platform level. Here's a CLI example:

```
az vm encryption enable -g ${RGNAME} --name ${VMNAME} --disk-encryption-keyvault "<your-unique-keyvault-name>"
```

18. Check the encryption information from the portal:

LUN ⓘ	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (Mbps)	Encryption ⓘ
0	vmresizely_datadisk0	Standard HDD	5	500	60	SSE with PMK & ADE
1	vmresizely_datadisk1	Standard HDD	5	500	60	SSE with PMK & ADE
2	vmresizely_datadisk2	Standard HDD	5	500	60	SSE with PMK & ADE
3	vmresizely_datadisk3	Standard HDD	5	500	60	SSE with PMK & ADE

After the encryption settings are updated, you can delete the new LV. Also delete the entry from the `/etc/fstab` and `/etc/crypttab` that you created.

```
[root@azurevm]# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Thu Jun 20 18:47:49 2019
#
# Accessible filesystems, by reference, are maintained under '/dev/disk'
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info
#
UUID=52e42afc-2da4-4391-bbc0-92f2alb45c62 /          xfs      defaults        0 0
UUID=a8fbef35-f5e4-4515-bb7e-c408c118d4c0 /boot       xfs      defaults        0 0
/dev/mapper/55b4af2-a160-426b-9bd7-588af6c46e9b /data1v01 ext4 defaults,nofail 0 0
/dev/mapper/78b259a1-956b-4f3a-8e6b-c05bf5040bc5 /data1v02 ext4 defaults,nofail 0 0
/dev/mapper/70abfc58-b0fd-441b-8b77-f86c9249af5e /data1v03 ext4 defaults,nofail 0 0
/dev/mapper/e5dee6bc-0f22-4579-b6c0-6103a9aba86c /data1v04 ext4 defaults,nofail 0 0
/dev/mapper/9157c7b5-8bad-4a84-84e2-0ca2bfffce4 /data1v05 ext4 defaults,nofail 0 0
LABEL=BERV040VOLUME /mnt/azure_bek_disk auto defaults,discard,nofail 0 0
[root@azurevm]#
[root@azurevm]#
[root@azurevm]# cat /etc/crypttab
55b4af2-a160-426b-9bd7-588af6c46e9b /dev/mapper/datavg-data1v01 /mnt/azure_bek_disk/LinuxPassPhraseFileName_1_0 luks,nofail
78b259a1-956b-4f3a-8e6b-c05bf5040bc5 /dev/mapper/datavg-data1v02 /mnt/azure_bek_disk/LinuxPassPhraseFileName_1_0 luks,nofail
70abfc58-b0fd-441b-8b77-f86c9249af5e /dev/mapper/datavg-data1v03 /mnt/azure_bek_disk/LinuxPassPhraseFileName_1_0 luks,nofail
e5dee6bc-0f22-4579-b6c0-6103a9aba86c /dev/mapper/datavg-data1v04 /mnt/azure_bek_disk/LinuxPassPhraseFileName_1_2 luks,nofail
9157c7b5-8bad-4a84-84e2-0ca2bfffce4 /dev/mapper/datavg-data1v05 /mnt/azure_bek_disk/LinuxPassPhraseFileName_1_3 luks,nofail
[root@azurevm]#
```

Follow these steps to finish cleaning up:

1. Unmount the LV:

```
umount /mountpoint
```

2. Close the encrypted layer of the volume:

```
cryptsetup luksClose /dev/vgname/lvname
```

3. Delete the LV:

```
lvremove /dev/vgname/lvname
```

### Extend a traditional LVM volume by resizing an existing PV

In some scenarios, your limitations might require you to resize an existing disk. Here's how:

1. Identify your encrypted disks:

```
ls -l /dev/disk/azure/scsi1/
```

```
[root@vmresizelv ~]# ls -l /dev/disk/azure/scsi1/
total 0
lrwxrwxrwx. 1 root root 12 Sep 19 00:21 lun0 -> ../../../../../sdd
lrwxrwxrwx. 1 root root 12 Sep 19 00:21 lun1 -> ../../../../../sde
lrwxrwxrwx. 1 root root 12 Sep 19 00:21 lun2 -> ../../../../../sdf
lrwxrwxrwx. 1 root root 12 Sep 19 00:21 lun3 -> ../../../../../sdg
```

```
lsblk -fs
```

```
[root@vmresizelv ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
fd0           2:0    1   4K  0 disk
sda           8:0    0 32G  0 disk
└─sda1        8:1    0 500M 0 part  /boot
               8:2    0 31.5G 0 part  /
sdb           8:16   0 32G  0 disk
└─sdb1        8:17   0 32G  0 part  /mnt/resource
sdc           8:32   0 48M  0 disk
└─sdc1        8:33   0 46M  0 part  /mnt/azure_bek_disk
sdd           8:48   0 5G   0 disk
└─datavg-datalv01 253:0  0 20G  0 lvm
  └─bb92534b-47a8-40da-adff-c4e9c0f643f0 253:1  0 20G  0 [crypt] /datalv01
sde           8:64   0 5G   0 disk
└─datavg-datalv01 253:0  0 20G  0 lvm
  └─bb92534b-47a8-40da-adff-c4e9c0f643f0 253:1  0 20G  0 [crypt] /datalv01
sdf           8:80   0 5G   0 disk
└─datavg-datalv01 253:0  0 20G  0 lvm
  └─bb92534b-47a8-40da-adff-c4e9c0f643f0 253:1  0 20G  0 [crypt] /datalv01
sdg           8:96   0 5G   0 disk
└─datavg-datalv01 253:0  0 20G  0 lvm
  └─bb92534b-47a8-40da-adff-c4e9c0f643f0 253:1  0 20G  0 [crypt] /datalv01
sr0          11:0   1 1024M 0 rom
```

2. Check the PV information:

```
pvs
```

```
[root@vmresizelv ~]# pvs
PV          VG      Fmt  Attr PSize  PFree
/dev/sdd    datavg lvm2 a--  <5.00g  0
/dev/sde    datavg lvm2 a--  <5.00g  0
/dev/sdf    datavg lvm2 a--  <5.00g  0
/dev/sdg    datavg lvm2 a--  <5.00g  0
```

The results in the image show that all of the space on all of the PVs is currently used.

3. Check the VG information:

```
vgs
vgdisplay -v vgname
```

```

[root@vmresizelv ~]# vgs
  VG      #PV #LV #SN Attr   VSize  VFree
  datavg    4   1   0 wz--n- 19.98g  0
[root@vmresizelv ~]# vgdisplay datavg
--- Volume group ---
VG Name           datavg
System ID
Format          lvm2
Metadata Areas   4
Metadata Sequence No 16
VG Access        read/write
VG Status        resizable
MAX LV          0
Cur LV          1
Open LV          1
Max PV          0
Cur PV          4
Act PV          4
VG Size         19.98 GiB
PE Size         4.00 MiB
Total PE        5116
Alloc PE / Size 5116 / 19.98 GiB
Free PE / Size  0 / 0
VG UUID         jvc1Kz-FXLd-dNjC-14q7-k85I-Pp8z-bMqFLY

```

4. Check the disk sizes. You can use `fdisk` or `lsblk` to list the drive sizes.

```

for disk in `ls -l /dev/disk/azure/scsi1/* | awk -F/ '{print $NF}'` ; do echo "fdisk -l /dev/${disk}"
| grep ^Disk "; done | bash

lsblk -o "NAME,SIZE"

```

Disk	<code>/dev/sdd</code> :	5368 MB	, 5368709120 bytes, 10485760 sectors
Disk	<code>/dev/sde</code> :	5368 MB	, 5368709120 bytes, 10485760 sectors
Disk	<code>/dev/sdf</code> :	5368 MB	, 5368709120 bytes, 10485760 sectors
Disk	<code>/dev/sdg</code> :	5368 MB	, 5368709120 bytes, 10485760 sectors

Here we identified which PVs are associated with which LVs by using `lsblk -fs`. You can identify the associations by running `lvdisplay`.

```

lvdisplay --maps VG/LV
lvdisplay --maps datavg/data1v1

```

```
[root@vmresizelv ~]# lvdisplay --maps datavg/datalv01
--- Logical volume ---
LV Path                  /dev/datavg/datalv01
LV Name                 datalv01
VG Name                 datavg
LV UUID                 oiTd6A-PTqY-AbzV-Kd0J-0891-dlMP-jBUJUC
LV Write Access          read/write
LV Creation host, time  vmresizelv, 2020-09-19 00:21:25 +0000
LV Status                available
# open                   1
LV Size                  19.98 GiB
Current LE               5116
Segments                 4
Allocation               inherit
Read ahead sectors       auto
- currently set to      8192
Block device             253:0

--- Segments ---
Logical extents 0 to 1278:
  Type        linear
  Physical volume /dev/sdd
  Physical extents 0 to 1278

Logical extents 1279 to 2557:
  Type        linear
  Physical volume /dev/sde
  Physical extents 0 to 1278

Logical extents 2558 to 3836:
  Type        linear
  Physical volume /dev/sdf
  Physical extents 0 to 1278

Logical extents 3837 to 5115:
  Type        linear
  Physical volume /dev/sdg
  Physical extents 0 to 1278
```

In this case, all four data drives are part of the same VG and a single LV. Your configuration might differ.

#### 5. Check the current file system utilization:

```
df -h /datalvm*
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/bb92534b-47a8-40da-adff-c4e9c0f643f0	20G	20G	0	100%	/datalv01

#### 6. Resize the data disks by following the instructions in [Expand an Azure managed disk](#). You can use the portal, the CLI, or PowerShell.

##### **IMPORTANT**

You can't resize virtual disks while the VM is running. Deallocate your VM for this step.

#### 7. Start the VM and check the new sizes by using `fdisk`.

```
for disk in `ls -l /dev/disk/azure/scsi1/* | awk -F/ '{print $NF}'` ; do echo "fdisk -l /dev/${disk}" | grep ^Disk ; done | bash
```

```
lsblk -o "NAME,SIZE"
```

```
Disk /dev/sdd: 21.5 GB, 21474836480 bytes, 41943040 sectors  
Disk /dev/sde: 5368 MB, 5368709120 bytes, 10485760 sectors  
Disk /dev/sdf: 5368 MB, 5368709120 bytes, 10485760 sectors  
Disk /dev/sdg: 5368 MB, 5368709120 bytes, 10485760 sectors
```

In this case, `/dev/sdd` was resized from 5 G to 20 G.

8. Check the current PV size:

```
pvdisplay /dev/resizeddisk
```

```
[root@vmresizelv ~]# pvdisplay /dev/sdd  
--- Physical volume ---  
PV Name /dev/sdd  
VG Name datavg  
PV Size 5.00 GiB / not usable 4.00 MiB  
Allocatable yes (but full)  
PE Size 4.00 MiB  
Total PE 1279  
Free PE 0  
Allocated PE 1279  
PV UUID gy2E19-BHhh-1Hy1-MGJg-P6tu-iSeh-vYpO8U
```

Even though the disk was resized, the PV still has the previous size.

9. Resize the PV:

```
pvresize /dev/resizeddisk
```

```
[root@vmresizelv ~]# pvresize /dev/sdd  
Physical volume "/dev/sdd" changed  
1 physical volume(s) resized / 0 physical volume(s) not resized
```

10. Check the PV size:

```
pvdisplay /dev/resizeddisk
```

```
[root@vmresizelv ~]# pvdisplay /dev/sdd  
--- Physical volume ---  
PV Name /dev/sdd  
VG Name datavg  
PV Size <20.00 GiB / not usable 3.00 MiB  
Allocatable yes  
PE Size 4.00 MiB  
Total PE 5119  
Free PE 3840  
Allocated PE 1279  
PV UUID gy2E19-BHhh-1Hy1-MGJg-P6tu-iSeh-vYpO8U
```

Apply the same procedure for all of the disks that you want to resize.

11. Check the VG information.

```
vgdisplay vgname
```

```
[root@vmresizelv ~]# vgdisplay datavg
--- Volume group ---
VG Name          datavg
System ID        lvm2
Format           lvm2
Metadata Areas   4
Metadata Sequence No 17
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV           1
Open LV          1
Max PV           0
Cur PV           4
Act PV           4
VG Size          34.98 GiB
PE Size          4.00 MiB
Total PE         8956
Alloc PE / Size  5116 / 19.98 GiB
Free  PE / Size  3840 / 15.00 GiB
VG UUID          jvc1Kz-FXLd-dNjC-14q7-k85I-Pp8z-bMqFLY
```

The VG now has enough space to be allocated to the LVs.

## 12. Resize the LV:

```
lvresize -r -L +5G vgname/lvname
lvresize -r -l +100%FREE /dev/datavg/datalv01
```

```
[root@vmresizelv ~]# lvresize -r -L +5G /dev/datavg/datalv01
Size of logical volume datavg/datalv01 changed from 19.98 GiB (5116 extents) to 24.98 GiB (6396 extents).
Logical volume datavg/datalv01 successfully resized.
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/bb92534b-47a8-40da-adff-c4e9c0f643f0 is mounted on /datalv01; on-line resizing required
old_desc_blocks = 3, new_desc_blocks = 4
The filesystem on /dev/mapper/bb92534b-47a8-40da-adff-c4e9c0f643f0 is now 6548992 blocks long.
```

## 13. Check the size of the file system:

```
df -h /datalv01
```

```
[root@vmresizelv ~]# df -h /datalv01/
Filesystem              Size  Used Avail Use% Mounted on
/dev/mapper/bb92534b-47a8-40da-adff-c4e9c0f643f0    35G   20G   14G  60% /datalv01
```

### Extend an LVM-on-crypt volume by adding a new PV

You can also extend an LVM-on-crypt volume by adding a new PV. This method closely follows the steps in [Configure LVM and RAID on encrypted devices](#). See the sections that explain how to add a new disk and set it up in an LVM-on-crypt configuration.

You can use this method to add space to an existing LV. Or you can create new VGs or LVs.

## 1. Verify the current size of your VG:

```
vgdisplay vgname
```

```
[root@vmresizelv ~]# vgdisplay datavg
--- Volume group ---
VG Name           datavg
System ID
Format           lvm2
Metadata Areas   3
Metadata Sequence No 9
VG Access        read/write
VG Status        resizable
MAX LV           0
Cur LV           4
Open LV           4
Max PV           0
Cur PV           3
Act PV           3
VG Size          <14.99 GiB
PE Size          4.00 MiB
Total PE         3837
Alloc PE / Size  3837 / <14.99 GiB
Free  PE / Size  0 / 0
VG UUID          4vKd5r-Q8wn-OVwP-vBiX-c9TH-xWzO-SdUVVM
```

2. Verify the size of the file system and LV that you want to expand:

```
lvdisplay /dev/vgname/lvname
```

```
[root@vmresizelv ~]# lvdisplay /dev/datavg/datalv01
--- Logical volume ---
LV Path           /dev/datavg/datalv01
LV Name           datalv01
VG Name           datavg
LV UUID           K8zZjy-brpp-Lx74-atfR-lTVu-h26Q-urRi7z
LV Write Access   read/write
LV Creation host, time vmresizelv, 2020-09-21 14:44:27 +0000
LV Status         available
# open            1
LV Size           1.00 GiB
Current LE        256
Segments          1
Allocation        inherit
Read ahead sectors auto
- currently set to 8192
Block device      253:4
```

```
df -h mountpoint
```

```
[root@vmresizelv ~]# df -h /datalv01
Filesystem           Size  Used Avail Use% Mounted on
/dev/mapper/datavg-datalv01  976M  960M    0 100% /datalv01
```

3. Add a new data disk to the VM and identify it.

Before you add the new disk, check the disks:

```
fdisk -l | egrep '^Disk /"
```

```
[root@vmresizelv ~]# fdisk -l | egrep '^Disk /"
Disk /dev/sdc: 50 MB, 50331648 bytes, 98304 sectors
Disk /dev/sdf: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sdg: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sde: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sdb: 34.4 GB, 34359738368 bytes, 67108864 sectors
Disk /dev/sda: 34.4 GB, 34359738368 bytes, 67108864 sectors
Disk /dev/mapper/resourcecrypt: 34.4 GB, 34356527104 bytes, 67102592 sectors
Disk /dev/mapper/e19fcfd77-e974-4e5c-a874-e78e4b6d2f48: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/49de5df0-1c65-48bf-809c-588805eda921: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/8a33198f-fa7e-491f-a9c6-e9d0a1d25d57: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/datavg-datalv01: 1073 MB, 1073741824 bytes, 2097152 sectors
Disk /dev/mapper/datavg-datalv02: 6442 MB, 6442450944 bytes, 12582912 sectors
Disk /dev/mapper/datavg-datalv03: 4294 MB, 4294967296 bytes, 8388608 sectors
Disk /dev/mapper/datavg-datalv04: 4282 MB, 4282384384 bytes, 8364032 sectors
```

Here's another way to check the disks before you add the new disk:

```
lsblk
```

```
[root@vmresizelv ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
fd0            2:0    1   4K  0 disk
sda            8:0    0 32G  0 disk
└─sda1         8:1    0 500M 0 part  /boot
  └─sda2         8:2    0 31.5G 0 part  /
sdb            8:16   0 32G  0 disk
└─sdb1         8:17   0 32G  0 part
  └─resourcecrypt  253:0  0 32G  0 crypt /mnt/resource
sdc            8:32   0 48M  0 disk
└─sdc1         8:33   0 47M  0 part  /mnt/azure_bek_disk
└─sde          8:64   0 5G  0 disk
  └─e19fcfd77-e974-4e5c-a874-e78e4b6d2f48 253:1  0 5G  0 crypt
    ├─datavg-datalv01  253:4  0 1G  0 lvm   /datalv01
    ├─datavg-datalv02  253:5  0 6G  0 lvm   /datalv02
    ├─datavg-datalv03  253:6  0 4G  0 lvm   /datalv03
    └─datavg-datalv04  253:7  0 3G  0 lvm   /datalv04
└─sdf          8:80   0 5G  0 disk
  └─49de5df0-1c65-48bf-809c-588805eda921 253:2  0 5G  0 crypt
    ├─datavg-datalv02  253:5  0 6G  0 lvm   /datalv02
    └─datavg-datalv04  253:7  0 3G  0 lvm   /datalv04
└─sdg          8:96   0 5G  0 disk
  └─8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 253:3  0 5G  0 crypt
    ├─datavg-datalv03  253:6  0 4G  0 lvm   /datalv03
    └─datavg-datalv04  253:7  0 3G  0 lvm   /datalv04
```

To add the new disk, you can use PowerShell, the Azure CLI, or the Azure portal. For more information, see [Attach a data disk to a Linux VM](#).

The kernel name scheme applies to the newly added device. A new drive is normally assigned the next available letter. In this case, the added disk is `sdd`.

#### 4. Check the disks to make sure the new disk has been added:

```
fdisk -l | egrep '^Disk /"
```

```
[root@vmresizelv ~]# fdisk -l | egrep '^Disk /"
Disk /dev/sdc: 50 MB, 50331648 bytes, 98304 sectors
Disk /dev/sdf: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sdg: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sde: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/sdb: 34.4 GB, 34359738368 bytes, 67108864 sectors
Disk /dev/sda: 34.4 GB, 34359738368 bytes, 67108864 sectors
Disk /dev/mapper/resourcecrypt: 34.4 GB, 34356527104 bytes, 67102592 sectors
Disk /dev/mapper/e19fcfd77-e974-4e5c-a874-e78e4b6d2f48: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/49de5df0-1c65-48bf-809c-588805eda921: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/8a33198f-fa7e-491f-a9c6-e9d0a1d25d57: 5366 MB, 5366611968 bytes, 10481664 sectors
Disk /dev/mapper/datavg-datalv01: 1073 MB, 1073741824 bytes, 2097152 sectors
Disk /dev/mapper/datavg-datalv02: 6442 MB, 6442450944 bytes, 12582912 sectors
Disk /dev/mapper/datavg-datalv03: 4294 MB, 4294967296 bytes, 8388608 sectors
Disk /dev/mapper/datavg-datalv04: 4282 MB, 4282384384 bytes, 8364032 sectors
Disk /dev/sdd: 5368 MB, 5368709120 bytes, 10485760 sectors
```

```
lsblk
```

```
[root@vmresizelv ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0           2:0    1   4K  0 disk 
sda           8:0    0  32G  0 disk 
└─sda1        8:1    0 500M  0 part /boot
└─sda2        8:2    0 31.5G 0 part /
sdb           8:16   0   32G  0 disk 
└─sdb1        8:17   0   32G  0 part 
  └─resourcecrypt 253:0   0   32G  0 crypt /mnt/resource
sdc           8:32   0   48M  0 disk 
└─sdc1        8:33   0   47M  0 part /mnt/azure_bek_disk
└─sdd         8:48   0     5G  0 disk 
  └─sde        8:64   0     5G  0 disk 
    └─e19fc77-e974-4e5c-a874-e78e4b6d2f48 253:1   0     5G  0 crypt
      ├─datavg-datalv01 253:4   0     1G  0 lvm   /datalv01
      ├─datavg-datalv02 253:5   0     6G  0 lvm   /datalv02
      ├─datavg-datalv03 253:6   0     4G  0 lvm   /datalv03
      └─datavg-datalv04 253:7   0     4G  0 lvm   /datalv04
sdf           8:80   0     5G  0 disk 
└─49de5df0-1c65-48bf-809c-588805eda921 253:2   0     5G  0 crypt
  ├─datavg-datalv02 253:5   0     6G  0 lvm   /datalv02
  └─datavg-datalv04 253:7   0     4G  0 lvm   /datalv04
sdg           8:96   0     5G  0 disk 
└─8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 253:3   0     5G  0 crypt
  ├─datavg-datalv03 253:6   0     4G  0 lvm   /datalv03
  └─datavg-datalv04 253:7   0     4G  0 lvm   /datalv04
```

5. Create a file system on top of the recently added disk. Match the disk to the linked devices on `/dev/disk/azure/scsi1/`.

```
ls -la /dev/disk/azure/scsi1/
```

```
[root@vmresizelv ~]# ls -la /dev/disk/azure/scsi1/*
lrwxrwxrwx. 1 root root 12 Sep 21 14:32 /dev/disk/azure/scsi1/lun1 -> ../../../../../../sde
lrwxrwxrwx. 1 root root 12 Sep 21 14:32 /dev/disk/azure/scsi1/lun2 -> ../../../../../../sdf
lrwxrwxrwx. 1 root root 12 Sep 21 14:32 /dev/disk/azure/scsi1/lun3 -> ../../../../../../sdg
lrwxrwxrwx. 1 root root 12 Sep 21 16:39 /dev/disk/azure/scsi1/lun4 -> ../../../../../../sdd
```

```
mkfs.ext4 /dev/disk/azure/scsi1/${disk}
```

```
[root@vmresizelv ~]# mkfs.ext4 /dev/disk/azure/scsi1/lun4
mke2fs 1.42.9 (28-Dec-2013)
/dev/disk/azure/scsi1/lun4 is entire device, not just one partition!
Proceed anyway? (y,n) y
Discarding device blocks: done
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
327680 inodes, 1310720 blocks
65536 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=1342177280
40 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
      32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

6. Create a temporary mount point for the new added disk:

```
newmount=/data4  
mkdir ${newmount}
```

7. Add the recently created file system to `/etc/fstab`.

```
blkid /dev/disk/azure/scsi1/lun4| awk -F\" '{print "UUID=\"$2\" \"$newmount\" \"$4\" defaults,nofail 0  
0\""}' >> /etc/fstab
```

8. Mount the newly created file system:

```
mount -a
```

9. Verify that the new file system is mounted:

```
df -h
```

```
[root@vmresizelv ~]# df -h  
Filesystem           Size  Used Avail Use% Mounted on  
/dev/sda2            32G  2.7G  29G  9% /  
devtmpfs             7.9G   0    7.9G  0% /dev  
tmpfs               7.9G   0    7.9G  0% /dev/shm  
tmpfs               7.9G  18M  7.9G  1% /run  
tmpfs               7.9G   0    7.9G  0% /sys/fs/cgroup  
/dev/sda1            497M 100M  397M 21% /boot  
tmpfs               1.6G   0    1.6G  0% /run/user/1000  
/dev/sdcl             43M  4.5K  43M  1% /mnt/azure_bek_disk  
/dev/mapper/resourcecrypt  32G  49M  30G  1% /mnt/resource  
tmpfs               1.6G   0    1.6G  0% /run/user/995  
/dev/mapper/datavg-datalv01  976M 960M   0 100% /datalv01  
/dev/mapper/datavg-datalv02  5.9G  5.9G   0 100% /datalv02  
/dev/mapper/datavg-datalv03  4.0G  3.9G   0 100% /datalv03  
/dev/mapper/datavg-datalv04  3.9G  4.0M  3.8G  1% /datalv04  
/dev/sdd              4.8G  20M  4.6G  1% /data4
```

```
lsblk
```

```
[root@vmresizelv ~]# lsblk  
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT  
fd0        2:0     1   4K  0 disk  
sda        8:0     0  32G  0 disk  
└─sda1     8:1     0 500M  0 part /boot  
        └─sda2     8:2     0 31.5G 0 part /  
sdb        8:16    0   32G  0 disk  
└─sdb1     8:17    0   32G  0 part  
    └─resourcecrypt 253:0   0 32G  0 crypt /mnt/resource  
sdc        8:32    0   48M  0 disk  
└─sdcl     8:33    0   47M  0 part /mnt/azure_bek_disk  
sdd        8:48    0   5G  0 disk /data4  
sde        8:64    0   5G  0 disk  
└─e19fc77-e974-4e5c-a874-e78e4b6d2f48 253:1   0   5G  0 crypt  
    ├─datavg-datalv01   253:4   0   1G  0 lvm  /datalv01  
    ├─datavg-datalv02   253:5   0   6G  0 lvm  /datalv02  
    ├─datavg-datalv03   253:6   0   4G  0 lvm  /datalv03  
    └─datavg-datalv04   253:7   0   4G  0 lvm  /datalv04  
sdf        8:80    0   5G  0 disk  
└─49de5df0-1c65-48bf-809c-588805eda921 253:2   0   5G  0 crypt  
    ├─datavg-datalv02   253:5   0   6G  0 lvm  /datalv02  
    └─datavg-datalv04   253:7   0   4G  0 lvm  /datalv04  
sdg        8:96    0   5G  0 disk  
└─8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 253:3   0   5G  0 crypt  
    ├─datavg-datalv03   253:6   0   4G  0 lvm  /datalv03  
    └─datavg-datalv04   253:7   0   4G  0 lvm  /datalv04
```

10. Restart the encryption that you previously started for data drives.

**TIP**

For LVM-on-crypt, we recommend that you use `EncryptFormatAll`. Otherwise, you might see a double encryption while you set additional disks.

For more information, see [Configure LVM and RAID on encrypted devices](#).

Here's an example:

```
az vm encryption enable \
--resource-group ${RGNAME} \
--name ${VMNAME} \
--disk-encryption-keyvault ${KEYVAULTNAME} \
--key-encryption-key ${KEYNAME} \
--key-encryption-keyvault ${KEYVAULTNAME} \
--volume-type "DATA" \
--encrypt-format-all \
-o table
```

When the encryption finishes, you see a crypt layer on the newly added disk:

```
lsblk
```

```
[root@vmresizelv ~]# lsblk
NAME          MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
fd0           2:0     1   4K  0 disk 
sda           8:0     0   32G  0 disk 
└─sda1        8:1     0   500M 0 part /boot
      └─sda2        8:2     0   31.5G 0 part /
sdb           8:16    0   32G  0 disk 
└─sdb1        8:17    0   32G  0 part 
      └─resourcecrypt 253:0    0   32G  0 crypt /mnt/resource
sdc           8:32    0   48M  0 disk 
└─sdc1        8:33    0   47M  0 part /mnt/azure_bek_disk
sdd           8:48    0   5G   0 disk 
└─8673c519-c5e4-44af-9b0f-fe286c2375b6 253:8    0   5G   0 crypt /data4
sde           8:64    0   5G   0 disk 
└─e19fc77-e974-4e5c-a874-e78e4b6d2f48 253:1    0   5G   0 crypt
    ├─datavg-datalv01 253:4    0   1G   0 lvm   /datalv01
    ├─datavg-datalv02 253:5    0   6G   0 lvm   /datalv02
    ├─datavg-datalv03 253:6    0   4G   0 lvm   /datalv03
    └─datavg-datalv04 253:7    0   4G   0 lvm   /datalv04
sdf           8:80    0   5G   0 disk 
└─49de5df0-1c65-48bf-809c-588805eda921 253:2    0   5G   0 crypt
    ├─datavg-datalv02 253:5    0   6G   0 lvm   /datalv02
    └─datavg-datalv04 253:7    0   4G   0 lvm   /datalv04
sdg           8:96    0   5G   0 disk 
└─8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 253:3    0   5G   0 crypt
    ├─datavg-datalv03 253:6    0   4G   0 lvm   /datalv03
    └─datavg-datalv04 253:7    0   4G   0 lvm   /datalv04
```

11. Unmount the encrypted layer of the new disk:

```
umount ${newmount}
```

12. Check the current PV information:

```
pvs
```

```
[root@vmresizelv ~]# pvs
PV              VG      Fmt  Attr PSize  PFree
/dev/mapper/49de5df0-1c65-48bf-809c-588805eda921 datavg lvm2 a-- <5.00g  0
/dev/mapper/8a33198f-fa7e-491f-a9c6-e9d0a1d25d57 datavg lvm2 a-- <5.00g  0
/dev/mapper/e19fc77-e974-4e5c-a874-e78e4b6d2f48 datavg lvm2 a-- <5.00g  0
```

13. Create a PV on top of the encrypted layer of the disk. Take the device name from the previous `lsblk` command. Add a `/dev/` mapper in front of the device name to create the PV:

```
pvcreate /dev/mapper/mapperdevicename
```

```
[root@vmresizelv ~]# pvcreate /dev/mapper/8673c519-c5e4-44af-9b0f-fe286c2375b6
WARNING: ext4 signature detected on /dev/mapper/8673c519-c5e4-44af-9b0f-fe286c2375b6 at offset 1080. Wipe it? [y/n]: y
Wiping ext4 signature on /dev/mapper/8673c519-c5e4-44af-9b0f-fe286c2375b6.
Physical volume "/dev/mapper/8673c519-c5e4-44af-9b0f-fe286c2375b6" successfully created.
```

You see a warning about wiping the current `ext4 fs` signature. This warning is expected. Answer this question with `y`.

14. Verify that the new PV was added to the LVM configuration:

```
pvs
```

```
[root@vmresizelv ~]# pvs
PV              VG      Fmt  Attr PSize  PFree
/dev/mapper/49de5df0-1c65-48bf-809c-588805eda921  datavg lvm2 a-- <5.00g    0
[redacted]
[redacted]
[redacted]
```

15. Add the new PV to the VG that you need to increase.

```
vgextend vgname /dev/mapper/nameofthenewpv
```

```
[root@vmresizelv ~]# vgextend datavg /dev/mapper/8673c519-c5e4-44af-9b0f-fe286c2375b6
Volume group "datavg" successfully extended
```

16. Verify the new size and free space of the VG:

```
vgdisplay vgname
```

```
[root@vmresizelv ~]# vgdisplay datavg
--- Volume group ---
VG Name          datavg
System ID
Format          lvm2
Metadata Areas   4
Metadata Sequence No 10
VG Access        read/write
VG Status        resizable
MAX LV
Cur LV
Open LV
Max PV
Cur PV
Act PV
VG Size         19.98 GiB
PE Size          4.00 MiB
Total PE         5116
Alloc PE / Size  3837 / <14.99 GiB
Free PE / Size   1279 / <5.00 GiB
VG UUID          4vKd5r-Q8wn-OVwP-vBiX-c9TH-xWzO-SdUVVM
```

Note the increase of the `Total PE` count and the `Free PE / Size`.

17. Increase the size of the LV and the file system. Use the `-r` option on `lvextend`. In this example, we're adding the total available space in the VG to the given LV.

```
lvextend -r -l +100%FREE /dev/vgname/lvname
```

```
[root@vmresizelv ~]# lvextend -r -l +100%FREE datavg/datalv01
  Size of logical volume datavg/datalv01 changed from 1.00 GiB (256 extents) to <6.00 GiB (1535 extents).
  Logical volume datavg/datalv01 successfully resized.
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/datavg-datalv01 is mounted on /datalv01; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/mapper/datavg-datalv01 is now 1571840 blocks long.
```

Follow the next steps to verify your changes.

1. Verify the size of the LV:

```
lvdisplay /dev/vgname/lvname
```

```
[root@vmresizelv ~]# lvdisplay /dev/datavg/datalv01
--- Logical volume ---
LV Path          /dev/datavg/datalv01
LV Name          datalv01
VG Name          datavg
LV UUID          K8zZjy-brpp-Lx74-atfR-lTVu-h26Q-urRi7z
LV Write Access  read/write
LV Creation host, time  vmresizelv, 2020-09-21 14:44:27 +0000
LV Status        available
# open           1
LV Size          <6.00 GiB
Current LE       1535
Segments         2
Allocation       inherit
Read ahead sectors auto
- currently set to 8192
Block device     253:4
```

2. Verify the new size of the file system:

```
df -h mountpoint
```

```
[root@vmresizelv ~]# df -h /datalv01/
Filesystem          Size  Used Avail Use% Mounted on
/dev/mapper/datavg-datalv01  5.9G  962M  4.7G 17% /datalv01
```

3. Verify that the LVM layer is on top of the encrypted layer:

```
lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└ sda1	8:1	0	500M	0	part	/boot
└ sda2	8:2	0	31.5G	0	part	/
sdb	8:16	0	32G	0	disk	
└ sdb1	8:17	0	32G	0	part	
└ resourcecrypt	253:0	0	32G	0	crypt	/mnt/resource
sdc	8:32	0	48M	0	disk	
└ sdc1	8:33	0	47M	0	part	/mnt/azure_bek_disk
sdd	8:48	0	5G	0	disk	
└ 8673c519-c5e4-44af-9b0f-fe286c2375b6	253:8	0	5G	0	crypt	
└ datavg-datalv01	253:4	0	6G	0	lvm	/datalv01
sde	8:64	0	5G	0	disk	
└ e19fc77-e974-4e5c-a874-e78e4b6d2f48	253:1	0	5G	0	crypt	
└ datavg-datalv01	253:4	0	6G	0	lvm	/datalv01
└ datavg-datalv02	253:5	0	6G	0	lvm	/datalv02
└ datavg-datalv03	253:6	0	4G	0	lvm	/datalv03
└ datavg-datalv04	253:7	0	4G	0	lvm	/datalv04
sdf	8:80	0	5G	0	disk	
└ 49de5df0-1c65-48bf-809c-588805eda921	253:2	0	5G	0	crypt	
└ datavg-datalv02	253:5	0	6G	0	lvm	/datalv02
└ datavg-datalv04	253:7	0	4G	0	lvm	/datalv04
sdg	8:96	0	5G	0	disk	
└ 8a33198f-fa7e-491f-a9c6-e9d0a1d25d57	253:3	0	5G	0	crypt	
└ datavg-datalv03	253:6	0	4G	0	lvm	/datalv03
└ datavg-datalv04	253:7	0	4G	0	lvm	/datalv04

If you use `lsblk` without options, then you see the mount points multiple times. The command sorts by device and LVs.

You might want to use `lsblk -fs`. In this command, `-fs` reverses the sort order so that the mount points are shown once. The disks are shown multiple times.

```
lsblk -fs
```

NAME	FSTYPE	LABEL	UUID	MOUNTPOINT
fd0				
sda	xfs		a8fbef35-ff5e-4515-bb7e-c408c118d4c0	/boot
└ sda1	xfs		52e42afc-2da4-4391-bbc0-92f2alb45c62	/
sdc	vfat	BEK VOLUME DCAC-0179		/mnt/azure_bek_disk
└ sdc1				
resourcecrypt	ext4		15e2c329-d4f3-4f44-a7de-6a4d8c0cf10e	/mnt/resource
└ sdb1	crypto_LUKS		10d7a858-082c-45f1-8fd6-b33fd9a0e83f	
└ sdb				
└ datavg-datalv01	ext4		6218e192-f1c3-47fa-be70-ccfe8267cffd	/datalv01
└ e19fc77-e974-4e5c-a874-e78e4b6d2f48	LVM2_member		02Sfy1-lnRK-OX2L-0WSZ-Hqb9-U9TH-EbAqUB	
└ sde	crypto_LUKS		98ae5085-aeb5-4025-a72f-7fbc875d427f	
└ 8673c519-c5e4-44af-9b0f-fe286c2375b6	LVM2_member		HPPc8-cfdj-Uvkt-ruow-bHXP-pRAF-cDRlp6	
└ sdd	crypto_LUKS		b0ffeb4-bb87-43fa-bafa-80d951945063	
└ datavg-datalv02	ext4		fec39096-4ac6-487b-b7b5-19796661d6b1	/datalv02
└ e19fc77-e974-4e5c-a874-e78e4b6d2f48	LVM2_member		02Sfy1-lnRK-OX2L-0WSZ-Hqb9-U9TH-EbAqUB	
└ sde	crypto_LUKS		98ae5085-aeb5-4025-a72f-7fbc875d427f	
└ 49de5df0-1c65-48bf-809c-588805eda921	LVM2_member		Hayij0-DL01-oS65-qHZ0-PcQJ-Br3v-I3Idye	
└ sdf	crypto_LUKS		f1754d5a-1858-4150-b38a-5c087aadb5e6	
└ datavg-datalv03	ext4		64f015a9-3e7b-4a67-922d-7cb68969f3dc	/datalv03
└ e19fc77-e974-4e5c-a874-e78e4b6d2f48	LVM2_member		02Sfy1-lnRK-OX2L-0WSZ-Hqb9-U9TH-EbAqUB	
└ sde	crypto_LUKS		98ae5085-aeb5-4025-a72f-7fbc875d427f	
└ 8a33198f-fa7e-491f-a9c6-e9d0a1d25d57	LVM2_member		I3zQkS-5Rnh-VdfK-MXwu-BhXk-8V23-DEjlsf	
└ sdd	crypto_LUKS		6b26ab4b-8091-49fe-9460-1d0d7dd8b4ac	
└ datavg-datalv04	ext4		dc20d93a-fc12-457e-8109-e73aced4ed31	/datalv04
└ e19fc77-e974-4e5c-a874-e78e4b6d2f48	LVM2_member		02Sfy1-lnRK-OX2L-0WSZ-Hqb9-U9TH-EbAqUB	
└ sde	crypto_LUKS		98ae5085-aeb5-4025-a72f-7fbc875d427f	
└ 49de5df0-1c65-48bf-809c-588805eda921	LVM2_member		Hayij0-DL01-oS65-qHZ0-PcQJ-Br3v-I3Idye	
└ sdf	crypto_LUKS		f1754d5a-1858-4150-b38a-5c087aadb5e6	
└ 8a33198f-fa7e-491f-a9c6-e9d0a1d25d57	LVM2_member		I3zQkS-5Rnh-VdfK-MXwu-BhXk-8V23-DEjlsf	
└ sda	crypto_LUKS		6b26ab4b-8091-49fe-9460-1d0d7dd8b4ac	

#### Extend an LVM on a crypt volume by resizing an existing PV

- Identify your encrypted disks:

```
lsblk
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:0	0	32G	0	disk	
└─sda1	8:1	0	500M	0	part	/boot
└─sda2	8:2	0	31.5G	0	part	/
sdb	8:16	0	32G	0	disk	
└─sdb1	8:17	0	32G	0	part	
└─resourcecrypt	253:0	0	32G	0	crypt	/mnt/resource
sdc	8:32	0	48M	0	disk	
└─sdc1	8:33	0	47M	0	part	/mnt/azure_bek_disk
sdd	8:48	0	2G	0	disk	
└─2c7d881a-6f89-4ef5-992a-944bdb26f4d8	253:1	0	2G	0	crypt	
└─datavg-datalv1	253:3	0	1.5G	0	lvm	/datalvml
└─datavg-datalv2	253:4	0	2.5G	0	lvm	/datalvml2
sde	8:64	0	2G	0	disk	
└─ba122ca3-1865-43c1-ba96-6d88e3ebcce1	253:2	0	2G	0	crypt	
└─datavg-datalv2	253:4	0	2.5G	0	lvm	/datalvml2

```
lsblk -s
```

NAME	MAJ:MIN	RM	SIZE	RO	TYPE	MOUNTPOINT
fd0	2:0	1	4K	0	disk	
sda	8:1	0	500M	0	part	/boot
└─sda1	8:0	0	32G	0	disk	
└─sda2	8:2	0	31.5G	0	part	/
sdb	8:0	0	32G	0	disk	
└─sdb1	8:17	0	32G	0	part	
└─resourcecrypt	253:0	0	32G	0	crypt	/mnt/resource
└─sdb1	8:16	0	32G	0	disk	
└─datavg-datalv1	253:3	0	1.5G	0	lvm	/datalvml
└─2c7d881a-6f89-4ef5-992a-944bdb26f4d8	253:1	0	2G	0	crypt	
└─sdd	8:48	0	2G	0	disk	
└─datavg-datalv2	253:4	0	2.5G	0	lvm	/datalvml2
└─2c7d881a-6f89-4ef5-992a-944bdb26f4d8	253:1	0	2G	0	crypt	
└─sdd	8:48	0	2G	0	disk	
└─ba122ca3-1865-43c1-ba96-6d88e3ebcce1	253:2	0	2G	0	crypt	
└─sde	8:64	0	2G	0	disk	

2. Check your PV information:

```
pvs
```

PV	VG	Fmt	Attr	PSize	PFree
/dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8	datavg	lvm2	a--	<2.00g	0
/dev/mapper/ba122ca3-1865-43c1-ba96-6d88e3ebcce1	datavg	lvm2	a--	<2.00g	0

3. Check your VG information:

```
vgs
```

```
[root@vm ]# vgdisplay datavg
--- Volume group ---
VG Name           datavg
System ID
Format           lvm2
Metadata Areas   2
Metadata Sequence No 3
VG Access        read/write
VG Status         resizable
MAX LV            0
Cur LV            2
Open LV           2
Max PV            0
Cur PV            2
Act PV            2
VG Size          3.99 GiB
PE Size          4.00 MiB
Total PE          1022
Alloc PE / Size  1022 / 3.99 GiB
Free PE / Size   0 / 0
VG UUID          24nwHl-raFN-igfO-0wk9-WTvc-sSTO-cXVYUX
```

4. Check your LV information:

```
lvs
```

```
[root@vm ]# lvs
LV      VG      Attr      LSize Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
datalv1  datavg -wi-ao---- 1.50g
datalv2  datavg -wi-ao---- 2.49g
```

5. Check the file system utilization:

```
df -h /mountpoint(s)
```

```
[root@vm ]# df -h /datalvm*
Filesystem              Size  Used Avail Use% Mounted on
/dev/mapper/datavg-datalv1 1.5G  1.5G    0 100% /datalvml
/dev/mapper/datavg-datalv2 2.4G  2.4G    0 100% /datalvm2
```

6. Check the sizes of your disks:

```
fdisk
fdisk -l | egrep ^"Disk /"
lsblk
```

```
[root@vm ]# fdisk -l /dev/sdd
Disk /dev/sdd: 2147 MB, 2147483648 bytes, 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

[root@vm ]# fdisk -l /dev/sde
Disk /dev/sde: 2147 MB, 2147483648 bytes, 4194304 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

7. Resize the data disk. You can use the portal, CLI, or PowerShell. For more information, see the disk-resize section in [Expand virtual hard disks on a Linux VM](#).

**IMPORTANT**

You can't resize virtual disks while the VM is running. Deallocate your VM for this step.

8. Check your disks sizes:

```
fdisk
fdisk -l | egrep '^Disk /'
lsblk
```

```
[root@vmresizelvm ~]# fdisk -l /dev/sdd
Disk /dev/sdd: 4294 MB, 4294967296 bytes, 8388608 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes

[root@vmresizelvm ~]# fdisk -l /dev/sde
Disk /dev/sde: 4294 MB, 4294967296 bytes, 8388608 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
```

In this case, both disks were resized from 2 GB to 4 GB. But the size of the file system, LV, and PV remain the same.

9. Check the current PV size. Remember that on LVM-on-crypt, the PV is the `/dev/mapper/` device, not the `/dev/sd*` device.

```
pvdisplay /dev/mapper/devicemappername
```

```
[root@vmresizelvm ~]# pvdisplay /dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8
--- Physical volume ---
PV Name           /dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8
VG Name           datavg
PV Size          <2.00 GiB / not usable 2.00 MiB
Allocatable       yes (but full)
PE Size          4.00 MiB
Total PE         511
Free PE          0
Allocated PE     511
PV UUID          30iBnM-fZgv-1sXe-FJo2-052x-wjxu-8Fb6Mw
```

10. Resize the PV:

```
pvresize /dev/mapper/devicemappername
```

```
[root@vmresizelvm ~]# [pvresize /dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8
Physical volume "/dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8" changed
[1 physical volume(s) resized] / 0 physical volume(s) not resized
```

11. Check the new PV size:

```
pvdisplay /dev/mapper/devicemappername
```

```
[root@vmresizelvm ~]# pvdisplay /dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8
--- Physical volume ---
PV Name           /dev/mapper/2c7d881a-6f89-4ef5-992a-944bdb26f4d8
VG Name           datavg
PV Size          <4.00 GiB / not usable 0
Allocatable       yes
PE Size          4.00 MiB
Total PE         1023
Free PE          512
Allocated PE     511
PV UUID          30iBnM-fZgv-1sXe-FJo2-052x-wjxu-8Fb6Mw
```

12. Resize the encrypted layer on the PV:

```
cryptsetup resize /dev/mapper/devicemappername
```

Apply the same procedure for all of the disks that you want to resize.

13. Check your VG information:

```
vgdisplay vgname
```

```
[root@vmresizelvm ~]# vgdisplay datavg
--- Volume group ---
VG Name           datavg
System ID        lvm2
Format           lvm2
Metadata Areas   2
Metadata Sequence No 4
VG Access        read/write
VG Status         resizable
MAX LV            0
Cur LV            2
Open LV            2
Max PV            0
Cur PV            2
Act PV            2
VG Size          5.99 GiB
PE Size          4.00 MiB
Total PE          1534
Alloc PE / Size  1022 / 3.99 GiB
Free  PE / Size  512 / 2.00 GiB
VG UUID          24nwHl-raFN-igfO-0wk9-WTvc-sSTO-cXVYUX
```

The VG now has enough space to be allocated to the LVs.

#### 14. Check the LV information:

```
lvdisplay vgname/lvname
```

```
[root@vmresizelvm ~]# lvdisplay datavg/datalv2
--- Logical volume ---
LV Path           /dev/datavg/datalv2
LV Name           datalv2
VG Name           datavg
LV UUID           iK3HYF-wYPX-XAs0-0WGB-Zite-DlaT-dFae2o
LV Write Access   read/write
LV Creation host, time  vmresizelvm, 2020-05-18 18:33:54 +0000
LV Status         available
# open            1
LV Size           2.49 GiB
Current LE        638
Segments          2
Allocation        inherit
Read ahead sectors auto
- currently set to 8192
Block device      253:3
```

#### 15. Check the file system utilization:

```
df -h /mountpoint
```

```
[root@vmresizelvm ~]# df -h /datalvm2/
Filesystem          Size  Used Avail Use% Mounted on
/dev/mapper/datavg-datalv2  2.4G  2.4G     0 100% /datalvm2
```

#### 16. Resize the LV:

```
lvresize -r -L +2G /dev/vgname/lvname
```

```
[root@vmresizelvm]# lvresize -r -L +2G /dev/datavg/datalv2
  Size of logical volume datavg/datalv2 changed from 2.50 GiB (640 extents) to 4.50 GiB (1152 extents).
  Logical volume datavg/datalv2 successfully resized.
resize2fs 1.42.9 (28-Dec-2013)
Filesystem at /dev/mapper/datavg-datalv2 is mounted on /datalvm2; on-line resizing required
old_desc_blocks = 1, new_desc_blocks = 1
The filesystem on /dev/mapper/datalv2 is now 1179648 blocks long.
```

Here we use the `-r` option to also resize the file system.

17. Check the LV information:

```
lvdisplay vgname/lvname
```

```
[root@vmresizelvm ~]# lvdisplay datavg/datalv2
--- Logical volume ---
  LV Path          /dev/datavg/datalv2
  LV Name          datalv2
  VG Name          datavg
  LV UUID          iK3HYF-wYPX-XAs0-0WGB-Zite-DlaT-dFae2o
  LV Write Access  read/write
  LV Creation host, time  vmresizelvm, 2020-05-18 18:33:54 +0000
  LV Status        available
  # open           1
  LV Size          4.49 GiB
  Current LE       1150
  Segments         3
  Allocation       inherit
  Read ahead sectors  auto
    - currently set to   8192
  Block device     253:3
```

18. Check the file system utilization:

```
df -h /mountpoint
```

```
[root@vmresizelvm ~]# df -h /datalvm2/
  Filesystem      Size  Used Avail Use% Mounted on
  /dev/mapper/datalv2  4.4G  2.4G  1.8G  58% /datalvm2
```

Apply the same resizing procedure to any other LV that requires it.

## Next steps

[Troubleshoot Azure Disk Encryption](#)

# Azure Disk Encryption for Linux VMs troubleshooting guide

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This guide is for IT professionals, information security analysts, and cloud administrators whose organizations use Azure Disk Encryption. This article is to help with troubleshooting disk-encryption-related problems.

Before taking any of the steps below, first ensure that the VMs you are attempting to encrypt are among the [supported VM sizes and operating systems](#), and that you have met all the prerequisites:

- [Additional requirements for VMs](#)
- [Networking requirements](#)
- [Encryption key storage requirements](#)

## Troubleshooting Linux OS disk encryption

Linux operating system (OS) disk encryption must unmount the OS drive before running it through the full disk encryption process. If it can't unmount the drive, an error message of "failed to unmount after ..." is likely to occur.

This error can occur when OS disk encryption is attempted on a VM with an environment that has been changed from the supported stock gallery image. Deviations from the supported image can interfere with the extension's ability to unmount the OS drive. Examples of deviations can include the following items:

- Customized images no longer match a supported file system or partitioning scheme.
- Large applications such as SAP, MongoDB, Apache Cassandra, and Docker aren't supported when they're installed and running in the OS before encryption. Azure Disk Encryption is unable to shut down these processes safely as required in preparation of the OS drive for disk encryption. If there are still active processes holding open file handles to the OS drive, the OS drive can't be unmounted, resulting in a failure to encrypt the OS drive.
- Custom scripts that run in close time proximity to the encryption being enabled, or if any other changes are being made on the VM during the encryption process. This conflict can happen when an Azure Resource Manager template defines multiple extensions to execute simultaneously, or when a custom script extension or other action runs simultaneously to disk encryption. Serializing and isolating such steps might resolve the issue.
- Security Enhanced Linux (SELinux) hasn't been disabled before enabling encryption, so the unmount step fails. SELinux can be reenabled after encryption is complete.
- The OS disk uses a Logical Volume Manager (LVM) scheme. Although limited LVM data disk support is available, an LVM OS disk isn't.
- Minimum memory requirements aren't met (7 GB is suggested for OS disk encryption).
- Data drives are recursively mounted under the /mnt/ directory, or each other (for example, /mnt/data1, /mnt/data2, /data3 + /data3/data4).

## Update the default kernel for Ubuntu 14.04 LTS

The Ubuntu 14.04 LTS image ships with a default kernel version of 4.4. This kernel version has a known issue in which Out of Memory Killer improperly terminates the dd command during the OS encryption process. This

bug has been fixed in the most recent Azure tuned Linux kernel. To avoid this error, prior to enabling encryption on the image, update to the [Azure tuned kernel 4.15](#) or later using the following commands:

```
sudo apt-get update  
sudo apt-get install linux-azure  
sudo reboot
```

After the VM has restarted into the new kernel, the new kernel version can be confirmed using:

```
uname -a
```

## Update the Azure Virtual Machine Agent and extension versions

Azure Disk Encryption operations may fail on virtual machine images using unsupported versions of the Azure Virtual Machine Agent. Linux images with agent versions earlier than 2.2.38 should be updated prior to enabling encryption. For more information, see [How to update the Azure Linux Agent on a VM](#) and [Minimum version support for virtual machine agents in Azure](#).

The correct version of the Microsoft.Azure.Security.AzureDiskEncryption or Microsoft.Azure.Security.AzureDiskEncryptionForLinux guest agent extension is also required. Extension versions are maintained and updated automatically by the platform when Azure Virtual Machine agent prerequisites are satisfied and a supported version of the virtual machine agent is used.

The Microsoft.OSTCExtensions.AzureDiskEncryptionForLinux extension has been deprecated and is no longer supported.

## Unable to encrypt Linux disks

In some cases, the Linux disk encryption appears to be stuck at "OS disk encryption started" and SSH is disabled. The encryption process can take between 3-16 hours to finish on a stock gallery image. If multi-terabyte-sized data disks are added, the process might take days.

The Linux OS disk encryption sequence unmounts the OS drive temporarily. It then performs block-by-block encryption of the entire OS disk, before it remounts it in its encrypted state. Linux Disk Encryption doesn't allow for concurrent use of the VM while the encryption is in progress. The performance characteristics of the VM can make a significant difference in the time required to complete encryption. These characteristics include the size of the disk and whether the storage account is standard or premium (SSD) storage.

While the OS drive is being encrypted, the VM enters a servicing state and disables SSH to prevent any disruption to the ongoing process. To check the encryption status, use the Azure PowerShell [Get-AzVmDiskEncryptionStatus](#) command, and check the **ProgressMessage** field. **ProgressMessage** will report a series of statuses as the data and OS disks are encrypted:

```

PS > Get-AzVMDiskEncryptionStatus -ResourceGroupName "MyResourceGroup" -VMName "myVM"

OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings :
ProgressMessage         : Transitioning

PS > Get-AzVMDiskEncryptionStatus -ResourceGroupName "MyResourceGroup" -VMName "myVM"

OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : Encryption succeeded for data volumes

PS > Get-AzVMDiskEncryptionStatus -ResourceGroupName "MyResourceGroup" -VMName "myVM"

OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : Provisioning succeeded

PS > Get-AzVMDiskEncryptionStatus -ResourceGroupName "MyResourceGroup" -VMName "myVM"

OsVolumeEncrypted      : EncryptionInProgress
DataVolumesEncrypted   : EncryptionInProgress
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : OS disk encryption started

```

The **ProgressMessage** will remain in **OS disk encryption started** for most of the encryption process. When encryption is complete and successful, **ProgressMessage** will return:

```

PS > Get-AzVMDiskEncryptionStatus -ResourceGroupName "MyResourceGroup" -VMName "myVM"

OsVolumeEncrypted      : Encrypted
DataVolumesEncrypted   : NotMounted
OsVolumeEncryptionSettings : Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage         : Encryption succeeded for all volumes

```

After this message is available, the encrypted OS drive is expected to be ready for use and the VM is ready to be used again.

If the boot information, the progress message, or an error reports that OS encryption has failed in the middle of this process, restore the VM to the snapshot or backup taken immediately before encryption. An example of a message is the "failed to unmount" error that is described in this guide.

Before reattempting encryption, reevaluate the characteristics of the VM and make sure that all of the prerequisites are satisfied.

## Troubleshooting Azure Disk Encryption behind a firewall

See [Disk Encryption on an isolated network](#)

## Troubleshooting encryption status

The portal may display a disk as encrypted even after it has been unencrypted within the VM. This can occur when low-level commands are used to directly unencrypt the disk from within the VM, instead of using the higher level Azure Disk Encryption management commands. The higher level commands not only unencrypt the disk from within the VM, but outside of the VM they also update important platform level encryption settings and extension settings associated with the VM. If these are not kept in alignment, the platform will not be able to

report encryption status or provision the VM properly.

To disable Azure Disk Encryption with PowerShell, use [Disable-AzVMDiskEncryption](#) followed by [Remove-AzVMDiskEncryptionExtension](#). Running Remove-AzVMDiskEncryptionExtension before the encryption is disabled will fail.

To disable Azure Disk Encryption with CLI, use [az vm encryption disable](#).

## Next steps

In this document, you learned more about some common problems in Azure Disk Encryption and how to troubleshoot those problems. For more information about this service and its capabilities, see the following articles:

- [Apply disk encryption in Microsoft Defender for Cloud](#)
- [Azure data encryption at rest](#)



# Upgrading the Azure Disk Encryption version

9/21/2022 • 2 minutes to read • [Edit Online](#)

The first version of Azure Disk Encryption (ADE) relied on Azure Active Directory (AAD) for authentication; the current version does not. We strongly encourage the use of the newest version.

## Determine ADE version

The versions of ADE in scope for migration are:

- **Windows:** 1.1.\* (ADE on the VM must be upgraded to 2.2)
- **Linux:** 0.1.\* (ADE on the VM must be upgraded to 1.2)

You can determine the version of ADE with which a VM was encrypted via Azure CLI, Azure PowerShell, or the Azure portal.

- [CLI](#)
- [PowerShell](#)
- [Portal](#)

To determine the ADE version, run the Azure CLI `az vm get-instance-view` command.

```
az vm get-instance-view --resource-group <ResourceGroupName> --name <VMName>
```

Locate the `AzureDiskEncryption` extension in the output and identify the version number from the "TypeHandlerVersion" field in the output.

## How to migrate

Migration from Azure Disk Encryption (with AAD) to Azure Disk Encryption (without AAD) is only available through Azure PowerShell. Ensure you have the latest version of Azure PowerShell and at least the [Azure PowerShell Az module version 5.9.0](#) installed .

To upgrade from Azure Disk Encryption (with AAD) to Azure Disk Encryption (without AAD), use the [Set-AzVMDiskEncryptionExtension](#) PowerShell cmdlet.

### WARNING

The `Set-AzVMDiskEncryptionExtension` cmdlet must only be used on VMs encrypted with Azure Disk Encryption (with AAD). Attempting to migrate an unencrypted VM, or a VM encrypted with Azure Disk Encryption (without AAD), will result in a terminal error.

```
Set-AzVMDiskEncryptionExtension -ResourceGroupName <resourceGroupName> -VMName <vmName> -Migrate
```

When the cmdlet prompts you for confirmation, enter "Y". The ADE version will be updated and the VM rebooted. The output will look similar to the following:

```

> Set-AzVMDiskEncryptionExtension -ResourceGroupName myResourceGroup -VMName myVM -Migrate

Update AzureDiskEncryption version?
This cmdlet updates Azure Disk Encryption version to single pass (Azure Disk Encryption without AAD). This
may reboot
the machine and takes 10-15 minutes to finish. Are you sure you want to continue?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
Azure Disk Encryption Extension Public Settings
"KeyVaultResourceId": /subscriptions/ea500758-3163-4849-bd2c-
3e50f06efa7a/resourceGroups/myResourceGroup/providers/Microsoft.KeyVault/vaults/myKeyVault
"SequenceVersion":
"MigrateFlag": Migrate
"KeyVaultURL": https://myKeyVault.vault.azure.net/
"AADCClientID": d29edf8c-3fcf-42e7-8410-9e39fdf0dd70
"KeyEncryptionKeyURL":
"KekVaultResourceId":
"EncryptionOperation": EnableEncryption
"AADCClientCertThumbprint":
"VolumeType":
"KeyEncryptionAlgorithm":

Running ADE extension (with AAD) for -Migrate..
ADE extension (with AAD) is now complete. Updating VM model..
Running ADE extension (without AAD) for -Migrate..
ADE extension (without AAD) is now complete. Clearing VM model..

RequestId IsSuccessStatusCode StatusCode ReasonPhrase
----- -----
True          OK   OK

```

#### **IMPORTANT**

The upgrade will take at least 10 - 15 minutes to complete. Do not cancel the cmdlet while the upgrade is in progress.  
Doing so puts the health of the VM at risk.

## Next steps

- [Azure Disk Encryption troubleshooting](#)

# Azure Disk Encryption with Azure Active Directory (AD) (previous release)

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure Active Directory (Azure AD) application parameter to enable VM disk encryption. With the new release, you're no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters by using the new release. For instructions on how to enable VM disk encryption by using the new release, see [Azure Disk Encryption for Linux VMs](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

This article provides supplements to [Azure Disk Encryption for Linux VMs](#) with additional requirements and prerequisites for Azure Disk Encryption with Azure AD (previous release).

The information in these sections remains the same:

- [Supported VMs and operating systems](#)
- [Additional VM requirements](#)

## Networking and Group Policy

To enable the Azure Disk Encryption feature by using the older AAD parameter syntax, the infrastructure as a service (IaaS) VMs must meet the following network endpoint configuration requirements:

- To get a token to connect to your key vault, the IaaS VM must be able to connect to an Azure AD endpoint, [login.microsoftonline.com].
- To write the encryption keys to your key vault, the IaaS VM must be able to connect to the key vault endpoint.
- The IaaS VM must be able to connect to an Azure storage endpoint that hosts the Azure extension repository and an Azure storage account that hosts the VHD files.
- If your security policy limits access from Azure VMs to the internet, you can resolve the preceding URI and configure a specific rule to allow outbound connectivity to the IPs. For more information, see [Azure Key Vault behind a firewall](#).
- On Windows, if TLS 1.0 is explicitly disabled and the .NET version isn't updated to 4.6 or higher, the following registry change enables Azure Disk Encryption to select the more recent TLS version:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\v4.0.30319]
"SystemDefaultTlsVersions"=dword:00000001
"SchUseStrongCrypto"=dword:00000001

[HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\.NETFramework\v4.0.30319]
"SystemDefaultTlsVersions"=dword:00000001
"SchUseStrongCrypto"=dword:00000001`
```

## Group Policy

- The Azure Disk Encryption solution uses the BitLocker external key protector for Windows IaaS VMs. For domain-joined VMs, don't push any Group Policies that enforce TPM protectors. For information about the Group Policy for the option **Allow BitLocker without a compatible TPM**, see [BitLocker Group](#)

## Policy reference.

- BitLocker policy on domain-joined virtual machines with a custom Group Policy must include the following setting: [Configure user storage of BitLocker recovery information -> Allow 256-bit recovery key](#). Azure Disk Encryption fails when custom Group Policy settings for BitLocker are incompatible. On machines that don't have the correct policy setting, apply the new policy, force the new policy to update (gpupdate.exe /force), and then restart if it's required.

## Encryption key storage requirements

Azure Disk Encryption requires Azure Key Vault to control and manage disk encryption keys and secrets. Your key vault and VMs must reside in the same Azure region and subscription.

For more information, see [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#).

## Next steps

- [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#)
- [Enable Azure Disk Encryption with Azure AD on Linux VMs \(previous release\)](#)
- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)

# Creating and configuring a key vault for Azure Disk Encryption with Azure AD (previous release) for Linux VMs

9/21/2022 • 15 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure AD application parameter to enable VM disk encryption. With the new release, you are no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters using the new release. To view instructions to enable VM disk encryption using the new release, see [Azure Disk Encryption](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

Azure Disk Encryption uses Azure Key Vault to control and manage disk encryption keys and secrets. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#).

Creating and configuring a key vault for use with Azure Disk Encryption with Azure AD (previous release) involves three steps:

1. Create a key vault.
2. Set up an Azure AD application and service principal.
3. Set the key vault access policy for the Azure AD app.
4. Set key vault advanced access policies.

You may also, if you wish, generate or import a key encryption key (KEK).

See the main [Creating and configuring a key vault for Azure Disk Encryption](#) article for steps on how to [Install tools and connect to Azure](#).

## NOTE

The steps in this article are automated in the [Azure Disk Encryption prerequisites CLI script](#) and [Azure Disk Encryption prerequisites PowerShell script](#).

## Create a key vault

Azure Disk Encryption is integrated with [Azure Key Vault](#) to help you control and manage the disk-encryption keys and secrets in your key vault subscription. You can create a key vault or use an existing one for Azure Disk Encryption. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#). You can use a Resource Manager template, Azure PowerShell, or the Azure CLI to create a key vault.

## WARNING

In order to make sure the encryption secrets don't cross regional boundaries, Azure Disk Encryption needs the Key Vault and the VMs to be co-located in the same region. Create and use a Key Vault that is in the same region as the VM to be encrypted.

## Create a key vault with PowerShell

You can create a key vault with Azure PowerShell using the [New-AzKeyVault](#) cmdlet. For additional cmdlets for Key Vault, see [Az.KeyVault](#).

1. Create a new resource group, if needed, with [New-AzResourceGroup](#). To list data center locations, use [Get-AzLocation](#).

```
# Get-AzLocation  
New-AzResourceGroup -Name 'MyKeyVaultResourceGroup' -Location 'East US'
```

2. Create a new key vault using [New-AzKeyVault](#)

```
New-AzKeyVault -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -Location  
'East US'
```

3. Note the **Vault Name**, **Resource Group Name**, **Resource ID**, **Vault URI**, and the **Object ID** that are returned for later use when you encrypt the disks.

## Create a key vault with Azure CLI

You can manage your key vault with Azure CLI using the `az keyvault` commands. To create a key vault, use [az keyvault create](#).

1. Create a new resource group, if needed, with [az group create](#). To list locations, use [az account list-locations](#)

```
# To list locations: az account list-locations --output table  
az group create -n "MyKeyVaultResourceGroup" -l "East US"
```

2. Create a new key vault using [az keyvault create](#).

```
az keyvault create --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --location "East  
US"
```

3. Note the **Vault Name** (name), **Resource Group Name**, **Resource ID** (ID), **Vault URI**, and the **Object ID** that are returned for use later.

## Create a key vault with a Resource Manager template

You can create a key vault by using the [Resource Manager template](#).

1. On the Azure quickstart template, click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, Key Vault name, Object ID, legal terms, and agreement, and then click **Purchase**.

## Set up an Azure AD app and service principal

When you need encryption to be enabled on a running VM in Azure, Azure Disk Encryption generates and writes the encryption keys to your key vault. Managing encryption keys in your key vault requires Azure AD authentication. Create an Azure AD application for this purpose. For authentication purposes, you can use either client secret-based authentication or [client certificate-based Azure AD authentication](#).

### Set up an Azure AD app and service principal with Azure PowerShell

To execute the following commands, get and use the [Azure AD PowerShell module](#).

1. Use the [New-AzADApplication](#) PowerShell cmdlet to create an Azure AD application.

MyApplicationHomePage and the MyApplicationUri can be any values you wish.

```
$aadClientSecret = "My AAD client secret"
$aadClientSecretSec = ConvertTo-SecureString -String $aadClientSecret -AsPlainText -Force
$azureAdApplication = New-AzADApplication -DisplayName "My Application Display Name" -HomePage
"https://MyApplicationHomePage" -IdentifierUris "https://MyApplicationUri" -Password
$aadClientSecretSec
$servicePrincipal = New-AzADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId -Role
Contributor
```

2. The \$azureAdApplication.ApplicationId is the Azure AD ClientID and the \$aadClientSecret is the client secret that you will use later to enable Azure Disk Encryption. Safeguard the Azure AD client secret appropriately. Running `$azureAdApplication.ApplicationId` will show you the ApplicationID.

### Set up an Azure AD app and service principal with Azure CLI

You can manage your service principals with Azure CLI using the `az ad sp` commands. For more information, see [Create an Azure service principal](#).

1. Create a new service principal.

```
az ad sp create-for-rbac --name "ServicePrincipalName" --password "My-AAD-client-secret" --role
Contributor --scopes /subscriptions/<subscription_id>
```

2. The appId returned is the Azure AD ClientID used in other commands. It's also the SPN you'll use for az keyvault set-policy. The password is the client secret that you should use later to enable Azure Disk Encryption. Safeguard the Azure AD client secret appropriately.

### Set up an Azure AD app and service principal though the Azure portal

Use the steps from the [Use portal to create an Azure Active Directory application and service principal that can access resources](#) article to create an Azure AD application. Each step listed below will take you directly to the article section to complete.

1. [Verify required permissions](#)
2. [Create an Azure Active Directory application](#)
  - You can use any name and sign-on URL you would like when creating the application.
3. [Get the application ID and the authentication key.](#)
  - The authentication key is the client secret and is used as the AadClientSecret for Set-AzVMDiskEncryptionExtension.
    - The authentication key is used by the application as a credential to sign in to Azure AD. In the Azure portal, this secret is called keys, but has no relation to key vaults. Secure this secret appropriately.
  - The application ID will be used later as the AadClientId for Set-AzVMDiskEncryptionExtension and as the ServicePrincipalName for Set-AzKeyVaultAccessPolicy.

## Set the key vault access policy for the Azure AD app

To write encryption secrets to a specified Key Vault, Azure Disk Encryption needs the Client ID and the Client Secret of the Azure Active Directory application that has permissions to write secrets to the Key Vault.

#### NOTE

Azure Disk Encryption requires you to configure the following access policies to your Azure AD client application: *WrapKey* and *Set* permissions.

## Set the key vault access policy for the Azure AD app with Azure PowerShell

Your Azure AD application needs rights to access the keys or secrets in the vault. Use the [Set-AzKeyVaultAccessPolicy](#) cmdlet to grant permissions to the application, using the client ID (which was generated when the application was registered) as the *-ServicePrincipalName* parameter value. To learn more, see the blog post [Azure Key Vault - Step by Step](#).

1. Set the key vault access policy for the AD application with PowerShell.

```
$keyVaultName = 'MySecureVault'  
$aadClientID = 'MyAadAppClientID'  
$KVRGname = 'MyKeyVaultResourceGroup'  
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -  
PermissionsToKeys 'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname
```

## Set the key vault access policy for the Azure AD app with Azure CLI

Use [az keyvault set-policy](#) to set the access policy. For more information, see [Manage Key Vault using CLI 2.0](#).

Give the service principal you created via the Azure CLI access to get secrets and wrap keys with the following command:

```
az keyvault set-policy --name "MySecureVault" --spn "<spn created with CLI/the Azure AD ClientID>" --key-permissions wrapKey --secret-permissions set
```

## Set the key vault access policy for the Azure AD app with the portal

1. Open the resource group with your key vault.
2. Select your key vault, go to **Access Policies**, then click **Add new**.
3. Under **Select principal**, search for the Azure AD application you created and select it.
4. For **Key permissions**, check **Wrap Key** under **Cryptographic Operations**.
5. For **Secret permissions**, check **Set** under **Secret Management Operations**.
6. Click **OK** to save the access policy.

Add new permissions    -    □    X

Add a new access policy - PREVIEW

\* Select principal  
vmencrypt >

Configure from template (optional)

Key permissions  
1 selected >

Secret permissions  
1 selected >

Authorized application ⓘ  
None selected

Key permissions

All Key Operations

All

Key Management Operations

Get

List

Update

Create

Import

Delete

Backup

Restore

Cryptographic Operations

Decrypt

Encrypt

UnwrapKey

WrapKey

Verify

Sign

Add new permissions    -    □    X

Add a new access policy - PREVIEW

\* Select principal  
vmencrypt >

Configure from template (optional)

Key permissions  
1 selected >

Secret permissions  
1 selected >

Authorized application ⓘ  
None selected

Secret permissions

All Secret Operations

All

Secret Management Operations

Get

List

Set

Delete

# Set key vault advanced access policies

The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the VM for booting and decrypting the volumes. Enable disk encryption on the key vault or deployments will fail.

## Set key vault advanced access policies with Azure PowerShell

Use the key vault PowerShell cmdlet [Set-AzKeyVaultAccessPolicy](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** EnabledForDiskEncryption is required for Azure Disk encryption.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForDiskEncryption
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForDeployment
```

- **Enable Key Vault for template deployment, if needed:** Enables Azure Resource Manager to get secrets from this key vault when this key vault is referenced in a template deployment.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForTemplateDeployment
```

## Set key vault advanced access policies using the Azure CLI

Use [az keyvault update](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** Enabled-for-disk-encryption is required.

```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
disk-encryption "true"
```

- **Enable Key Vault for deployment, if needed:** Allow Virtual Machines to retrieve certificates stored as secrets from the vault.

```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
deployment "true"
```

- **Enable Key Vault for template deployment, if needed:** Allow Resource Manager to retrieve secrets from the vault.

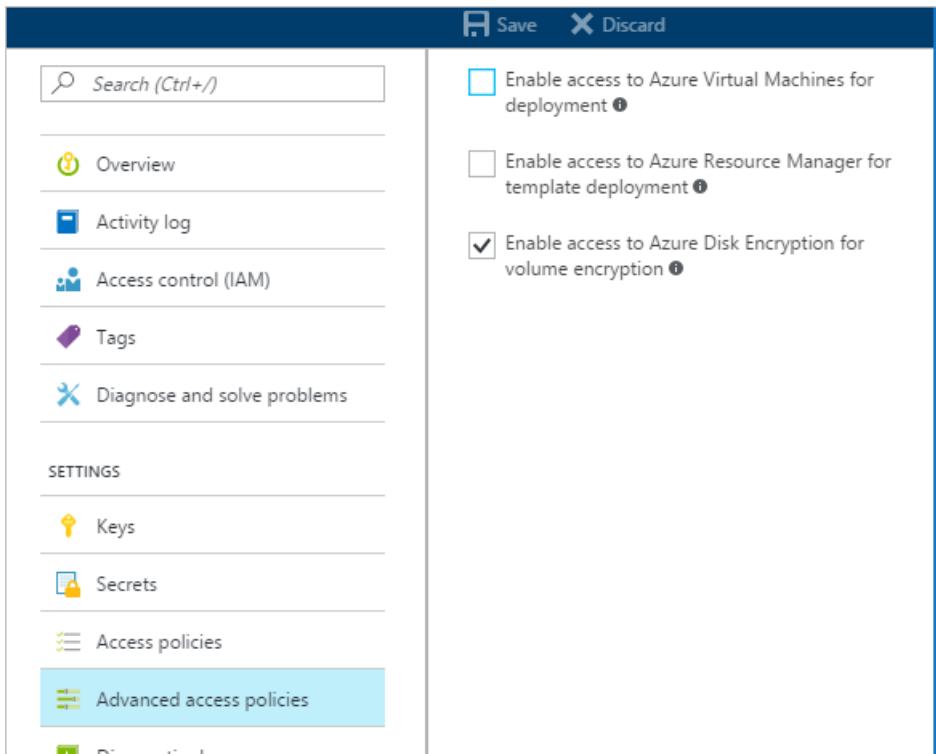
```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
template-deployment "true"
```

## Set key vault advanced access policies through the Azure portal

1. Select your keyvault, go to Access Policies, and Click to show advanced access policies.
2. Select the box labeled Enable access to Azure Disk Encryption for volume encryption.

3. Select **Enable access to Azure Virtual Machines for deployment** and/or **Enable Access to Azure Resource Manager for template deployment**, if needed.

4. Click **Save**.



## Set up a key encryption key (optional)

If you want to use a key encryption key (KEK) for an additional layer of security for encryption keys, add a KEK to your key vault. Use the [Add-AzKeyVaultKey](#) cmdlet to create a key encryption key in the key vault. You can also import a KEK from your on-premises key management HSM. For more information, see [Key Vault Documentation](#). When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault.

- When generating keys, use an RSA key type. Azure Disk Encryption does not yet support using Elliptic Curve keys.
- Your key vault secret and KEK URLs must be versioned. Azure enforces this restriction of versioning. For valid secret and KEK URLs, see the following examples:
  - Example of a valid secret URL:  
<https://contosovault.vault.azure.net/secrets/EncryptionSecretWithKek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - Example of a valid KEK URL:  
<https://contosovault.vault.azure.net/keys/diskencryptionkek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
- Azure Disk Encryption doesn't support specifying port numbers as part of key vault secrets and KEK URLs. For examples of non-supported and supported key vault URLs, see the following examples:
  - Unacceptable key vault URL  
<https://contosovault.vault.azure.net:443/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - Acceptable key vault URL:  
<https://contosovault.vault.azure.net/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

### Set up a key encryption key with Azure PowerShell

Before using the PowerShell script, you should be familiar with the Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment. This script creates all Azure Disk Encryption prerequisites and encrypts an existing IaaS VM, wrapping the disk encryption

key by using a key encryption key.

```
# Step 1: Create a new resource group and key vault in the same location.
# Fill in 'MyLocation', 'MyKeyVaultResourceGroup', and 'MySecureVault' with your values.
# Use Get-AzLocation to get available locations and use the DisplayName.
# To use an existing resource group, comment out the line for New-AzResourceGroup

$Loc = 'MyLocation';
$KVRGname = 'MyKeyVaultResourceGroup';
$keyVaultName = 'MySecureVault';
New-AzResourceGroup -Name $KVRGname -Location $Loc;
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc;
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$keyVaultResourceId = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname).ResourceId;
$diskEncryptionKeyVaultUrl = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname).VaultUri;

# Step 2: Create the AD application and service principal.
# Fill in 'MyAADClientSecret', "<My Application Display Name>", "<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish.

$aadClientSecret = 'MyAADClientSecret';
$aadClientSecretSec = ConvertTo-SecureString -String $aadClientSecret -AsPlainText -Force;
$azureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -Password $aadClientSecretSec
$servicePrincipal = New-AzADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId -Role Contributor;
$aadClientID = $azureAdApplication.ApplicationId;

#Step 3: Enable the vault for disk encryption and set the access policy for the Azure AD application.

Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -
EnabledForDiskEncryption;
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -
PermissionsToKeys 'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname;

#Step 4: Create a new key in the key vault with the Add-AzKeyVaultKey cmdlet.
# Fill in 'MyKeyEncryptionKey' with your value.

$keyEncryptionKeyName = 'MyKeyEncryptionKey';
Add-AzKeyVaultKey -VaultName $keyVaultName -Name $keyEncryptionKeyName -Destination 'Software';
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name $keyEncryptionKeyName).Key.kid;

#Step 5: Encrypt the disks of an existing IaaS VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM';
$VMRGName = 'MyVirtualMachineResourceGroup';
Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID $aadClientID -
AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -
DiskEncryptionKeyId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyId $keyVaultResourceId;
```

## Certificate-based authentication (optional)

If you would like to use certificate authentication, you can upload one to your key vault and deploy it to the client. Before using the PowerShell script, you should be familiar with the Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment.

```
# Fill in "MyKeyVaultResourceGroup", "MySecureVault", and 'MyLocation' ('My location' only if needed)
```

```

$KVRGname = 'MyKeyVaultResourceGroup'
$keyVaultName= 'MySecureVault'

# Create a key vault and set enabledForDiskEncryption property on it.
# Comment out the next three lines if you already have an existing key vault enabled for encryption. No
need to set 'My location' in this case.

$Loc = 'MyLocation'
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDiskEncryption

#Setting some variables with the key vault information
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname
$DiskEncryptionKeyVaultUrl = $keyVault.VaultUri
$keyVaultResourceId = $keyVault.ResourceId

# Create the Azure AD application and associate the certificate with it.
# Fill in "C:\certificates\mycert.pfx", "Password", "<My Application Display Name>", "
<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish

$CertPath = "C:\certificates\mycert.pfx"
$CertPassword = "Password"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath,
$CertPassword)
$CertValue = [System.Convert]::ToBase64String($cert.GetRawCertData())

$AzureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "
<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -CertValue $CertValue
$ServicePrincipal = New-AzADServicePrincipal -ApplicationId $AzureAdApplication.ApplicationId -Role
Contributor

$AADClientID = $AzureAdApplication.ApplicationId
$aadClientCertThumbprint= $cert.Thumbprint

Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -PermissionsToKeys
'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname

# Upload the pfx file to the key vault.
# Fill in "MyAADCert".

$keyVaultSecretName = "MyAADCert"
$fileContentBytes = get-content $CertPath -Encoding Byte
$fileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)
$jsonObject = @{
    "data" : "$filecontentencoded",
    "dataType" : "pfx",
    "password" : "$CertPassword"
}
"@

$jsonObjectBytes = [System.Text.Encoding]::UTF8.GetBytes($jsonObject)
$jsonEncoded = [System.Convert]::ToBase64String($jsonObjectBytes)

#Set the secret and set the key vault policy for -EnabledForDeployment

$secret = ConvertTo-SecureString -String $jsonEncoded -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName -SecretValue $secret
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDeployment

# Deploy the certificate to the VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM'
$VMRGName = 'MyVirtualMachineResourceGroup'
$CertUrl = (Get-AzKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName).Id
$SourceVaultId = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname).ResourceId
$VM = Get-AzVM -ResourceGroupName $VMRGName -Name $VMName

```

```
$VM = Add-AzVMSecret -VM $VM -SourceVaultId $SourceVaultId -CertificateStore "My" -CertificateUrl $CertUrl
Update-AzVM -VM $VM -ResourceGroupName $VMRGName

#Enable encryption on the VM using Azure AD client ID and the client certificate thumbprint

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -
AadClientCertThumbprint $AADClientCertThumbprint -DiskEncryptionKeyVaultUrl $DiskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $KeyVaultResourceId
```

## Certificate-based authentication and a KEK (optional)

If you would like to use certificate authentication and wrap the encryption key with a KEK, you can use the below script as an example. Before using the PowerShell script, you should be familiar with all of the previous Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment.

### IMPORTANT

Azure AD certificate-based authentication is currently not supported on Linux VMs.

```
# Fill in 'MyKeyVaultResourceGroup', 'MySecureVault', and 'MyLocation' (if needed)

$KVRGname = 'MyKeyVaultResourceGroup'
$keyVaultName= 'MySecureVault'

# Create a key vault and set enabledForDiskEncryption property on it.
# Comment out the next three lines if you already have an existing key vault enabled for encryption.

$Loc = 'MyLocation'
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDiskEncryption

# Create the Azure AD application and associate the certificate with it.
# Fill in "C:\certificates\mycert.pfx", "Password", "<My Application Display Name>", "
<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish

$CertPath = "C:\certificates\mycert.pfx"
$CertPassword = "Password"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath,
$CertPassword)
$CertValue = [System.Convert]::ToBase64String($cert.GetRawCertData())

$AzureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "
<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -CertValue $CertValue
$ServicePrincipal = New-AzADServicePrincipal -ApplicationId $AzureAdApplication.ApplicationId -Role
Contributor

$AADClientID = $AzureAdApplication.ApplicationId
$aadClientCertThumbprint= $cert.Thumbprint

## Give access for setting secrets and wrapping keys
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -PermissionsToKeys
'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname

# Upload the pfx file to the key vault.
# Fill in "MyAADCert".

$keyVaultSecretName = "MyAADCert"
$FileContentBytes = get-content $CertPath -Encoding Byte
$FileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)
$JSONObject = @"
{
    "name": "MyAADCert",
    "type": "AsymmetricX509Certificate",
    "value": $FileContentEncoded
}"
```

```

        "data" : "$filecontentencoded",
        "dataType" : "pfx",
        "password" : "$CertPassword"
    }
"""

$JSONObjectBytes = [System.Text.Encoding]::UTF8.GetBytes($jsonObject)
$JSONEncoded = [System.Convert]::ToBase64String($jsonObjectBytes)

#Set the secret and set the key vault policy for deployment

$Secret = ConvertTo-SecureString -String $JSONEncoded -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName $KeyVaultName -Name $KeyVaultSecretName -SecretValue $Secret
Set-AzKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $KVRGname -EnabledForDeployment

#Setting some variables with the key vault information and generating a KEK
# Fill in 'KEKName'

$KEKName = 'KEKName'
$keyVault = Get-AzKeyVault -VaultName $KeyVaultName -ResourceGroupName $KVRGname
$DiskEncryptionKeyVaultUrl = $keyVault.VaultUri
$keyVaultResourceId = $keyVault.ResourceId
$KEK = Add-AzKeyVaultKey -VaultName $KeyVaultName -Name $KEKName -Destination "Software"
$keyEncryptionKeyId = $KEK.Key.kid

# Deploy the certificate to the VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM';
$VMRGName = 'MyVirtualMachineResourceGroup';
$CertUrl = (Get-AzKeyVaultSecret -VaultName $KeyVaultName -Name $KeyVaultSecretName).Id
$SourceVaultId = (Get-AzKeyVault -VaultName $KeyVaultName -ResourceGroupName $KVRGName).ResourceId
$VM = Get-AzVM -ResourceGroupName $VMRGName -Name $VMName
$VM = Add-AzVMSecret -VM $VM -SourceVaultId $SourceVaultId -CertificateStore "My" -CertificateUrl $CertUrl
Update-AzVM -VM $VM -ResourceGroupName $VMRGName

#Enable encryption on the VM using Azure AD client ID and the client certificate thumbprint

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -
AadClientCertThumbprint $AADClientCertThumbprint -DiskEncryptionKeyVaultUrl $DiskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $KeyVaultResourceId -KeyEncryptionKeyId $keyEncryptionKeyId -
KeyEncryptionKeyId $KeyVaultResourceId

```

## Next steps

[Enable Azure Disk Encryption with Azure AD on Linux VMs \(previous release\)](#)

# Enable Azure Disk Encryption with Azure AD on Linux VMs (previous release)

9/21/2022 • 17 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure Active Directory (Azure AD) application parameter to enable VM disk encryption. With the new release, you're no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters by using the new release. For instructions on how to enable VM disk encryption by using the new release, see [Azure Disk Encryption for Linux VMs](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

You can enable many disk-encryption scenarios, and the steps might vary according to the scenario. The following sections cover the scenarios in greater detail for Linux infrastructure as a service (IaaS) VMs. You can only apply disk encryption to virtual machines of [supported VM sizes and operating systems](#). You must also meet the following prerequisites:

- [Additional requirements for VMs](#)
- [Networking and Group Policy](#)
- [Encryption key storage requirements](#)

Take a [snapshot](#), make a backup, or both before you encrypt the disks. Backups ensure that a recovery option is possible if an unexpected failure occurs during encryption. VMs with managed disks require a backup before encryption occurs. After a backup is made, you can use the `Set-AzVMDiskEncryptionExtension` cmdlet to encrypt managed disks by specifying the `-skipVmBackup` parameter. For more information about how to back up and restore encrypted VMs, see [Azure Backup](#).

## WARNING

- If you previously used [Azure Disk Encryption with the Azure AD app](#) to encrypt this VM, you must continue to use this option to encrypt your VM. You can't use [Azure Disk Encryption](#) on this encrypted VM because this isn't a supported scenario, which means switching away from the Azure AD application for this encrypted VM isn't supported yet.
- To make sure the encryption secrets don't cross regional boundaries, Azure Disk Encryption needs the key vault and the VMs to be co-located in the same region. Create and use a key vault that's in the same region as the VM to be encrypted.
- When you encrypt Linux OS volumes, the process can take a few hours. It's normal for Linux OS volumes to take longer than data volumes to encrypt.
- When you encrypt Linux OS volumes, the VM should be considered unavailable. We strongly recommend that you avoid SSH logins while the encryption is in progress to avoid blocking any open files that need to be accessed during the encryption process. To check progress, use the `Get-AzVMDiskEncryptionStatus` or `vm encryption show` commands. You can expect this process to take a few hours for a 30-GB OS volume, plus additional time for encrypting data volumes. Data volume encryption time is proportional to the size and quantity of the data volumes unless the **encrypt format all** option is used.
- Disabling encryption on Linux VMs is only supported for data volumes. It's not supported on data or OS volumes if the OS volume has been encrypted.

# Enable encryption on an existing or running IaaS Linux VM

In this scenario, you can enable encryption by using the Azure Resource Manager template, PowerShell cmdlets, or Azure CLI commands.

## IMPORTANT

It's mandatory to take a snapshot or back up a managed disk-based VM instance outside of and prior to enabling Azure Disk Encryption. You can take a snapshot of the managed disk from the Azure portal, or you can use [Azure Backup](#).

Backups ensure that a recovery option is possible in the case of any unexpected failure during encryption. After a backup is made, use the `Set-AzVMDiskEncryptionExtension` cmdlet to encrypt managed disks by specifying the `-skipVmBackup` parameter. The `Set-AzVMDiskEncryptionExtension` command fails against managed disk-based VMs until a backup is made and this parameter is specified.

Encrypting or disabling encryption might cause the VM to reboot.

## Enable encryption on an existing or running Linux VM by using the Azure CLI

You can enable disk encryption on your encrypted VHD by installing and using the [Azure CLI 2.0](#) command-line tool. You can use it in your browser with [Azure Cloud Shell](#), or you can install it on your local machine and use it in any PowerShell session. To enable encryption on existing or running IaaS Linux VMs in Azure, use the following CLI commands:

Use the `az vm encryption enable` command to enable encryption on a running IaaS virtual machine in Azure.

- **Encrypt a running VM by using a client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI/my Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --volume-type [All|OS|Data]
```

- **Encrypt a running VM by using KEK to wrap the client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI which is the Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type [All|OS|Data]
```

## NOTE

The syntax for the value of the `disk-encryption-keyvault` parameter is the full identifier string:  
`/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]`.

The syntax for the value of the `key-encryption-key` parameter is the full URI to the KEK as in: `https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]`.

- **Verify that the disks are encrypted:** To check on the encryption status of an IaaS VM, use the `az vm encryption show` command.

```
az vm encryption show --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup"
```

- **Disable encryption:** To disable encryption, use the `az vm encryption disable` command. Disabling encryption is only allowed on data volumes for Linux VMs.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --volume-type DATA
```

## Enable encryption on an existing or running Linux VM by using PowerShell

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet to enable encryption on a running IaaS virtual machine in Azure. Take a [snapshot](#) or make a backup of the VM with [Azure Backup](#) before the disks are encrypted. The `-skipVmBackup` parameter is already specified in the PowerShell scripts to encrypt a running Linux VM.

- **Encrypt a running VM by using a client secret:** The following script initializes your variables and runs the `Set-AzVMDiskEncryptionExtension` cmdlet. The resource group, VM, key vault, Azure AD app, and client secret should have already been created as prerequisites. Replace `MyVirtualMachineResourceGroup`, `MyKeyVaultResourceGroup`, `MySecureVM`, `MySecureVault`, `My-AAD-client-ID`, and `My-AAD-client-secret` with your values. Modify the `-VolumeType` parameter to specify which disks you're encrypting.

```
$VMRGName = 'MyVirtualMachineResourceGroup';
$KVRGname = 'MyKeyVaultResourceGroup';
$vmName = 'MySecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId -VolumeType '[All|OS|Data]' -SequenceVersion
$sequenceVersion -skipVmBackup;
```

- **Encrypt a running VM by using KEK to wrap the client secret:** Azure Disk Encryption lets you specify an existing key in your key vault to wrap disk encryption secrets that were generated while enabling encryption. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to the key vault. Modify the `-VolumeType` parameter to specify which disks you're encrypting.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $keyVaultResourceId -VolumeType '[All|OS|Data]' -SequenceVersion
$sequenceVersion -skipVmBackup;
```

**NOTE**

The syntax for the value of the disk-encryption-keyvault parameter is the full identifier string:  
/subscriptions/[subscription-id-guid]/resourceGroups/[KVresource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name].

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id].

- **Verify that the disks are encrypted:** To check on the encryption status of an IaaS VM, use the [Get-AzVmDiskEncryptionStatus](#) cmdlet.

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName MyVirtualMachineResourceGroup -VMName MySecureVM
```

- **Disable disk encryption:** To disable the encryption, use the [Disable-AzureRmVMDiskEncryption](#) cmdlet. Disabling encryption is only allowed on data volumes for Linux VMs.

```
Disable-AzVMDiskEncryption -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

### Enable encryption on an existing or running IaaS Linux VM with a template

You can enable disk encryption on an existing or running IaaS Linux VM in Azure by using the [Resource Manager template](#).

1. Select **Deploy to Azure** on the Azure quickstart template.
2. Select the subscription, resource group, resource group location, parameters, legal terms, and agreement. Select **Create** to enable encryption on the existing or running IaaS VM.

The following table lists Resource Manager template parameters for existing or running VMs that use an Azure AD client ID:

PARAMETER	DESCRIPTION
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to the key vault.
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to your key vault.
keyVaultName	Name of the key vault that the key should be uploaded to. You can get it by using the Azure CLI command <pre>az keyvault show --name "MySecureVault" --query KVresourceGroup</pre>
keyEncryptionKeyURL	URL of the key encryption key that's used to encrypt the generated key. This parameter is optional if you select <b>nokek</b> in the <b>UseExistingKek</b> drop-down list. If you select <b>kek</b> in the <b>UseExistingKek</b> drop-down list, you must enter the <i>keyEncryptionKeyURL</i> value.

PARAMETER	DESCRIPTION
volumeType	Type of volume that the encryption operation is performed on. Valid supported values are <i>OS</i> or <i>All</i> . (See supported Linux distributions and their versions for OS and data disks in the prerequisites section earlier.)
sequenceVersion	Sequence version of the BitLocker operation. Increment this version number every time a disk-encryption operation is performed on the same VM.
vmName	Name of the VM that the encryption operation is to be performed on.
passphrase	Type a strong passphrase as the data encryption key.

## Use the EncryptFormatAll feature for data disks on Linux IaaS VMs

The `EncryptFormatAll` parameter reduces the time for Linux data disks to be encrypted. Partitions that meet certain criteria are formatted (with their current file system). Then they're remounted back to where they were before command execution. If you want to exclude a data disk that meets the criteria, you can unmount it before you run the command.

After you run this command, any drives that were mounted previously are reformatted. Then the encryption layer starts on top of the now empty drive. When this option is selected, the temporary disk attached to the VM is also encrypted. If the ephemeral drive is reset, it's reformatted and re-encrypted for the VM by the Azure Disk Encryption solution at the next opportunity.

### WARNING

`EncryptFormatAll` shouldn't be used when there's needed data on a VM's data volumes. You can exclude disks from encryption by unmounting them. Try out the `EncryptFormatAll` parameter on a test VM first to understand the feature parameter and its implication before you try it on the production VM. The `EncryptFormatAll` option formats the data disk, so all the data on it will be lost. Before you proceed, verify that any disks you want to exclude are properly unmounted.

If you set this parameter while you update encryption settings, it might lead to a reboot before the actual encryption. In this case, you also want to remove the disk you don't want formatted from the `fstab` file. Similarly, you should add the partition you want encrypt-formatted to the `fstab` file before you initiate the encryption operation.

### EncryptFormatAll criteria

The parameter goes through all partitions and encrypts them as long as they meet *all* of the following criteria:

- Is not a root/OS/boot partition
- Is not already encrypted
- Is not a BEK volume
- Is not a RAID volume
- Is not an LVM volume
- Is mounted

Encrypt the disks that compose the RAID or LVM volume rather than the RAID or LVM volume.

### Use the EncryptFormatAll parameter with a template

To use the `EncryptFormatAll` option, use any preexisting Azure Resource Manager template that encrypts a Linux VM and change the `EncryptionOperation` field for the `AzureDiskEncryption` resource.

1. As an example, use the [Resource Manager template to encrypt a running Linux IaaS VM](#).
2. Select **Deploy to Azure** on the Azure quickstart template.
3. Change the **EncryptionOperation** field from **EnableEncryption** to **EnableEncryptionFormatAll**.
4. Select the subscription, resource group, resource group location, other parameters, legal terms, and agreement. Select **Create** to enable encryption on the existing or running IaaS VM.

## Use the **EncryptFormatAll** parameter with a **PowerShell cmdlet**

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet with the **EncryptFormatAll** parameter.

**Encrypt a running VM by using a client secret and **EncryptFormatAll**:** As an example, the following script initializes your variables and runs the [Set-AzVMDiskEncryptionExtension](#) cmdlet with the **EncryptFormatAll** parameter. The resource group, VM, key vault, Azure AD app, and client secret should have already been created as prerequisites. Replace **MyKeyVaultResourceGroup**, **MyVirtualMachineResourceGroup**, **MySecureVM**, **MySecureVault**, **My-AAD-client-ID**, and **My-AAD-client-secret** with your values.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID $aadClientID -
AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $keyVaultResourceId -EncryptFormatAll
```

## Use the **EncryptFormatAll** parameter with Logical Volume Manager (LVM)

We recommend an LVM-on-crypt setup. For all the following examples, replace the device-path and mountpoints with whatever suits your use case. This setup can be done as follows:

- Add the data disks that will compose the VM.
  - Format, mount, and add these disks to the fstab file.
1. Format the newly added disk. We use symlinks generated by Azure here. Using symlinks avoids problems related to device names changing. For more information, see [Troubleshoot device names problems](#).

```
mkfs -t ext4 /dev/disk/azure/scsi1/lun0
```

2. Mount the disks.

```
mount /dev/disk/azure/scsi1/lun0 /mnt/mountpoint
```

3. Add to fstab.

```
echo "/dev/disk/azure/scsi1/lun0 /mnt/mountpoint ext4 defaults,nofail 1 2" >> /etc/fstab
```

4. Run the [Set-AzVMDiskEncryptionExtension](#) PowerShell cmdlet with **-EncryptFormatAll** to encrypt these disks.

```
Set-AzVMDiskEncryptionExtension -ResourceGroupName "MySecureGroup" -VMName "MySecureVM" -  
DiskEncryptionKeyVaultUrl "https://mykeyvault.vault.azure.net/" -EncryptFormatAll
```

5. Set up LVM on top of these new disks. Note the encrypted drives are unlocked after the VM has finished booting. So, the LVM mounting will also have to be subsequently delayed.

## New IaaS VMs created from customer-encrypted VHD and encryption keys

In this scenario, you can enable encrypting by using the Resource Manager template, PowerShell cmdlets, or CLI commands. The following sections explain in greater detail the Resource Manager template and CLI commands.

Use the instructions in the appendix for preparing pre-encrypted images that can be used in Azure. After the image is created, you can use the steps in the next section to create an encrypted Azure VM.

- [Prepare a pre-encrypted Linux VHD](#)

### IMPORTANT

It's mandatory to take a snapshot or back up a managed disk-based VM instance outside of and prior to enabling Azure Disk Encryption. You can take a snapshot of the managed disk from the portal, or you can use [Azure Backup](#). Backups ensure that a recovery option is possible in the case of any unexpected failure during encryption. After a backup is made, use the `Set-AzVMDiskEncryptionExtension` cmdlet to encrypt managed disks by specifying the `-skipVmBackup` parameter. The `Set-AzVMDiskEncryptionExtension` command fails against managed disk-based VMs until a backup is made and this parameter is specified.

Encrypting or disabling encryption might cause the VM to reboot.

### Use Azure PowerShell to encrypt IaaS VMs with pre-encrypted VHDS

You can enable disk encryption on your encrypted VHD by using the PowerShell cmdlet `Set-AzVMOSDisk`. The following example gives you some common parameters.

```
$VirtualMachine = New-AzVMConfig -VMName "MySecureVM" -VMSize "Standard_A1"  
$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name "SecureOSDisk" -VhdUri "os.vhd" Caching ReadWrite  
-Windows -CreateOption "Attach" -DiskEncryptionKeyUrl  
"https://mytestvault.vault.azure.net/secrets/Test1/514ceb769c984379a7e0230bddaaaaaa" -  
DiskEncryptionKeyId "/subscriptions/00000000-0000-0000-0000-  
0000000000/resourceGroups/myresourcegroup/providers/Microsoft.KeyVault/vaults/mytestvault"  
New-AzVM -VM $VirtualMachine -ResourceGroupName "MyVirtualMachineResourceGroup"
```

## Enable encryption on a newly added data disk

You can add a new data disk by using [az vm disk attach](#) or [through the Azure portal](#). Before you can encrypt, you need to mount the newly attached data disk first. You must request encryption of the data drive because the drive will be unusable while encryption is in progress.

### Enable encryption on a newly added disk with the Azure CLI

If the VM was previously encrypted with "All," then the `--volume-type` parameter should remain All. All includes both OS and data disks. If the VM was previously encrypted with a volume type of "OS," then the `--volume-type` parameter should be changed to All so that both the OS and the new data disk will be included. If the VM was encrypted with only the volume type of "Data," then it can remain Data as demonstrated here. Adding and attaching a new data disk to a VM isn't sufficient preparation for encryption. The newly attached disk must also be formatted and properly mounted within the VM before you enable encryption. On Linux, the disk must be mounted in `/etc/fstab` with a [persistent block device name](#).

In contrast to PowerShell syntax, the CLI doesn't require you to provide a unique sequence version when you enable encryption. The CLI automatically generates and uses its own unique sequence version value.

- **Encrypt a running VM by using a client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI/my Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --volume-type "Data"
```

- **Encrypt a running VM by using KEK to wrap the client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI which is the Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type "Data"
```

## Enable encryption on a newly added disk with Azure PowerShell

When you use PowerShell to encrypt a new disk for Linux, a new sequence version needs to be specified. The sequence version has to be unique. The following script generates a GUID for the sequence version.

- **Encrypt a running VM by using a client secret:** The following script initializes your variables and runs the Set-AzVMDiskEncryptionExtension cmdlet. The resource group, VM, key vault, Azure AD app, and client secret should have already been created as prerequisites. Replace MyVirtualMachineResourceGroup, MyKeyVaultResourceGroup, MySecureVM, MySecureVault, My-AAD-client-ID, and My-AAD-client-secret with your values. The -VolumeType parameter is set to data disks and not the OS disk. If the VM was previously encrypted with a volume type of "OS" or "All," then the -VolumeType parameter should be changed to All so that both the OS and the new data disk will be included.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyId $keyVaultResourceId -VolumeType 'data' -SequenceVersion $sequenceVersion;
```

- **Encrypt a running VM by using KEK to wrap the client secret:** Azure Disk Encryption lets you specify an existing key in your key vault to wrap disk encryption secrets that were generated while enabling encryption. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to the key vault. The -VolumeType parameter is set to data disks and not the OS disk. If the VM was previously encrypted with a volume type of "OS" or "All," then the -VolumeType parameter should be changed to All so that both the OS and the new data disk will be included.

```

$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$KeyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $KeyVault.VaultUri;
$KeyVaultResourceId = $KeyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $KeyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $KeyVaultResourceId -VolumeType 'data' -SequenceVersion $sequenceVersion;

```

#### NOTE

The syntax for the value of the disk-encryption-keyvault parameter is the full identifier string: /subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name].

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id].

## Disable encryption for Linux VMs

You can disable encryption by using Azure PowerShell, the Azure CLI, or a Resource Manager template.

#### IMPORTANT

Disabling encryption with Azure Disk Encryption on Linux VMs is only supported for data volumes. It's not supported on data or OS volumes if the OS volume has been encrypted.

- Disable disk encryption with Azure PowerShell:** To disable encryption, use the [Disable-AzureRmVMDiskEncryption](#) cmdlet.

```
Disable-AzVMDiskEncryption -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName
'MySecureVM' [--volume-type {ALL, DATA, OS}]
```

- Disable encryption with the Azure CLI:** To disable encryption, use the [az vm encryption disable](#) command.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --
volume-type [ALL, DATA, OS]
```

- Disable encryption with a Resource Manager template:** To disable encryption, use the [Disable encryption on a running Linux VM](#) template.

- Select **Deploy to Azure**.
- Select the subscription, resource group, location, VM, legal terms, and agreement.
- Select **Purchase** to disable disk encryption on a running Windows VM.

## Next steps

- [Azure Disk Encryption for Linux overview](#)
- [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#)

# Azure Disk Encryption for Windows VMs

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Azure Disk Encryption helps protect and safeguard your data to meet your organizational security and compliance commitments. It uses the [BitLocker](#) feature of Windows to provide volume encryption for the OS and data disks of Azure virtual machines (VMs), and is integrated with [Azure Key Vault](#) to help you control and manage the disk encryption keys and secrets.

Azure Disk Encryption is zone resilient, the same way as Virtual Machines. For details, see [Azure Services that support Availability Zones](#).

If you use [Microsoft Defender for Cloud](#), you're alerted if you have VMs that aren't encrypted. The alerts show as High Severity and the recommendation is to encrypt these VMs.

VIRTUAL MACHINES RECOMMENDATIONS		TOTAL			
Missing disk encryption	2 of 2 VMs	<div style="width: 100%; background-color: red; height: 10px;"></div>			
Virtual machines					
NAME	ONBOARDING	SYSTEM UPDATES	ANTIMALWARE	BASELINE	DISK ENCRYPTION
ASC-VM1	✓	✓	✓	✓	!
ASC-VM2	✓	✓	✓	✓	!

## WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM. See [Azure Disk Encryption with Azure AD \(previous release\)](#) for details.
- Certain recommendations might increase data, network, or compute resource usage, resulting in additional license or subscription costs. You must have a valid active Azure subscription to create resources in Azure in the supported regions.
- Do not use BitLocker to manually decrypt a VM or disk that was encrypted through Azure Disk Encryption.

You can learn the fundamentals of Azure Disk Encryption for Windows in just a few minutes with the [Create and encrypt a Windows VM with Azure CLI quickstart](#) or the [Create and encrypt a Windows VM with Azure PowerShell quickstart](#).

## Supported VMs and operating systems

### Supported VMs

Windows VMs are available in a [range of sizes](#). Azure Disk Encryption is supported on Generation 1 and Generation 2 VMs. Azure Disk Encryption is also available for VMs with premium storage.

Azure Disk Encryption is not available on [Basic, A-series VMs](#), or on virtual machines with a less than 2 GB of memory. For more exceptions, see [Azure Disk Encryption: Unsupported scenarios](#).

## Supported operating systems

- Windows client: Windows 8 and later.
- Windows Server: Windows Server 2008 R2 and later.
- Windows 10 Enterprise multi-session.

### NOTE

Windows Server 2022 does not support an RSA 2048 bit key. For more details, see [FAQ: What size should I use for my key encryption key?](#)

Windows Server 2008 R2 requires the .NET Framework 4.5 to be installed for encryption; install it from Windows Update with the optional update Microsoft .NET Framework 4.5.2 for Windows Server 2008 R2 x64-based systems ([KB2901983](#)).

Windows Server 2012 R2 Core and Windows Server 2016 Core requires the bdehdcfg component to be installed on the VM for encryption.

## Networking requirements

To enable Azure Disk Encryption, the VMs must meet the following network endpoint configuration requirements:

- To get a token to connect to your key vault, the Windows VM must be able to connect to an Azure Active Directory endpoint, [login.microsoftonline.com].
- To write the encryption keys to your key vault, the Windows VM must be able to connect to the key vault endpoint.
- The Windows VM must be able to connect to an Azure storage endpoint that hosts the Azure extension repository and an Azure storage account that hosts the VHD files.
- If your security policy limits access from Azure VMs to the Internet, you can resolve the preceding URI and configure a specific rule to allow outbound connectivity to the IPs. For more information, see [Azure Key Vault behind a firewall](#).

## Group Policy requirements

Azure Disk Encryption uses the BitLocker external key protector for Windows VMs. For domain joined VMs, don't push any group policies that enforce TPM protectors. For information about the group policy for "Allow BitLocker without a compatible TPM," see [BitLocker Group Policy Reference](#).

BitLocker policy on domain joined virtual machines with custom group policy must include the following setting: [Configure user storage of BitLocker recovery information -> Allow 256-bit recovery key](#). Azure Disk Encryption will fail when custom group policy settings for BitLocker are incompatible. On machines that didn't have the correct policy setting, apply the new policy, and force the new policy to update (gpupdate.exe /force). Restarting may be required.

Microsoft Bitlocker Administration and Monitoring (MBAM) group policy features are not compatible with Azure Disk Encryption.

### WARNING

Azure Disk Encryption **does not store recovery keys**. If the [Interactive logon: Machine account lockout threshold](#) security setting is enabled, machines can only be recovered by providing a recovery key via the serial console. Instructions for ensuring the appropriate recovery policies are enabled can be found in the [Bitlocker recovery guide plan](#).

Azure Disk Encryption will fail if domain level group policy blocks the AES-CBC algorithm, which is used by BitLocker.

# Encryption key storage requirements

Azure Disk Encryption requires an Azure Key Vault to control and manage disk encryption keys and secrets. Your key vault and VMs must reside in the same Azure region and subscription.

For details, see [Creating and configuring a key vault for Azure Disk Encryption](#).

## Terminology

The following table defines some of the common terms used in Azure disk encryption documentation:

TERMINOLOGY	DEFINITION
Azure Key Vault	Key Vault is a cryptographic, key management service that's based on Federal Information Processing Standards (FIPS) validated hardware security modules. These standards help to safeguard your cryptographic keys and sensitive secrets. For more information, see the <a href="#">Azure Key Vault</a> documentation and <a href="#">Creating and configuring a key vault for Azure Disk Encryption</a> .
Azure CLI	The <a href="#">Azure CLI</a> is optimized for managing and administering Azure resources from the command line.
BitLocker	<a href="#">BitLocker</a> is an industry-recognized Windows volume encryption technology that's used to enable disk encryption on Windows VMs.
Key encryption key (KEK)	The asymmetric key (RSA 2048) that you can use to protect or wrap the secret. You can provide a hardware security module (HSM)-protected key or software-protected key. For more information, see the <a href="#">Azure Key Vault</a> documentation and <a href="#">Creating and configuring a key vault for Azure Disk Encryption</a> .
PowerShell cmdlets	For more information, see <a href="#">Azure PowerShell cmdlets</a> .

## Next steps

- [Quickstart - Create and encrypt a Windows VM with Azure CLI](#)
- [Quickstart - Create and encrypt a Windows VM with Azure PowerShell](#)
- [Azure Disk Encryption scenarios on Windows VMs](#)
- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)
- [Creating and configuring a key vault for Azure Disk Encryption](#)

# Quickstart: Create and encrypt a Windows VM with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

The Azure CLI is used to create and manage Azure resources from the command line or in scripts. This quickstart shows you how to use the Azure CLI to create and encrypt a Windows Server 2016 virtual machine (VM).

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This article requires version 2.0.30 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group

Create a resource group with the `az group create` command. An Azure resource group is a logical container into which Azure resources are deployed and managed. The following example creates a resource group named `myResourceGroup` in the `eastus` location:

```
az group create --name myResourceGroup --location eastus
```

## Create a virtual machine

Create a VM with `az vm create`. The following example creates a VM named `myVM`. This example uses `azureuser` for an administrative user name and `myPassword12` as the password.

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image win2016datacenter \
--admin-username azureuser \
--admin-password myPassword12
```

It takes a few minutes to create the VM and supporting resources. The following example output shows the VM create operation was successful.

```
{
  "fqdns": "",
  "id": "/subscriptions/<guid>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",
  "location": "eastus",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "52.174.34.95",
  "resourceGroup": "myResourceGroup"
}
```

## Create a Key Vault configured for encryption keys

Azure disk encryption stores its encryption key in an Azure Key Vault. Create a Key Vault with [az keyvault create](#). To enable the Key Vault to store encryption keys, use the --enabled-for-disk-encryption parameter.

### IMPORTANT

Each Key Vault must have a unique name. The following example creates a Key Vault named *myKV*, but you must name yours something different.

```
az keyvault create --name "myKV" --resource-group "myResourceGroup" --location eastus --enabled-for-disk-encryption
```

## Encrypt the virtual machine

Encrypt your VM with [az vm encryption](#), providing your unique Key Vault name to the --disk-encryption-keyvault parameter.

```
az vm encryption enable -g MyResourceGroup --name MyVM --disk-encryption-keyvault myKV
```

You can verify that encryption is enabled on your VM with [az vm show](#)

```
az vm encryption show --name MyVM -g MyResourceGroup
```

You will see the following in the returned output:

```
"EncryptionOperation": "EnableEncryption"
```

## Clean up resources

When no longer needed, you can use the [az group delete](#) command to remove the resource group, VM, and Key Vault.

```
az group delete --name myResourceGroup
```

## Next steps

In this quickstart, you created a virtual machine, created a Key Vault that was enable for encryption keys, and encrypted the VM. Advance to the next article to learn more about Azure Disk Encryption prerequisites for IaaS VMs.

[Azure Disk Encryption overview](#)

# Quickstart: Create and encrypt a Windows virtual machine in Azure with PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The Azure PowerShell module is used to create and manage Azure resources from the PowerShell command line or in scripts. This quickstart shows you how to use the Azure PowerShell module to create a Windows virtual machine (VM), create a Key Vault for the storage of encryption keys, and encrypt the VM.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Create a resource group

Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which Azure resources are deployed and managed:

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

## Create a virtual machine

Create an Azure virtual machine with [New-AzVM](#). You must supply credentials to the cmdlet.

```
$cred = Get-Credential  
  
New-AzVM -Name MyVm -Credential $cred -ResourceGroupName MyResourceGroup -Image win2016datacenter -Size Standard_D2S_V3
```

It will take a few minutes for your VM to be deployed.

## Create a Key Vault configured for encryption keys

Azure disk encryption stores its encryption key in an Azure Key Vault. Create a Key Vault with [New-AzKeyVault](#). To enable the Key Vault to store encryption keys, use the `-EnabledForDiskEncryption` parameter.

### IMPORTANT

Each Key Vault must have a unique name. The following example creates a Key Vault named *myKV*, but you must name yours something different.

```
New-AzKeyVault -name MyKV -ResourceGroupName myResourceGroup -Location EastUS -EnabledForDiskEncryption
```

## Encrypt the virtual machine

Encrypt your VM with [Set-AzVmDiskEncryptionExtension](#).

`Set-AzVmDiskEncryptionExtension` requires some values from your Key Vault object. You can obtain these values by passing the unique name of your key vault to [Get-AzKeyVault](#).

```
$KeyVault = Get-AzKeyVault -VaultName MyKV -ResourceGroupName MyResourceGroup

Set-AzVMDiskEncryptionExtension -ResourceGroupName MyResourceGroup -VMName MyVM -DiskEncryptionKeyVaultUrl
$KeyVault.VaultUri -DiskEncryptionKeyId $KeyVault.ResourceId
```

After a few minutes the process will return the following:

RequestId	IsSuccess	Status	Code	ReasonPhrase
	True	OK	OK	

You can verify the encryption process by running [Get-AzVmDiskEncryptionStatus](#).

```
Get-AzVmDiskEncryptionStatus -VMName MyVM -ResourceGroupName MyResourceGroup
```

When encryption is enabled, you will see the following in the returned output:

OsVolumeEncrypted	:	Encrypted
DataVolumesEncrypted	:	NoDiskFound
OsVolumeEncryptionSettings	:	Microsoft.Azure.Management.Compute.Models.DiskEncryptionSettings
ProgressMessage	:	Provisioning succeeded

## Clean up resources

When no longer needed, you can use the [Remove-AzResourceGroup](#) cmdlet to remove the resource group, VM, and all related resources:

```
Remove-AzResourceGroup -Name "myResourceGroup"
```

## Next steps

In this quickstart, you created a virtual machine, created a Key Vault that was enable for encryption keys, and encrypted the VM. Advance to the next article to learn more about Azure Disk Encryption prerequisites for IaaS VMs.

[Azure Disk Encryption overview](#)

# Quickstart: Create and encrypt a Windows virtual machine with the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

Azure virtual machines (VMs) can be created through the Azure portal. The Azure portal is a browser-based user interface to create VMs and their associated resources. In this quickstart you will use the Azure portal to deploy a Windows virtual machine, create a key vault for the storage of encryption keys, and encrypt the VM.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Sign in to Azure

Sign in to the [Azure portal](#).

## Create a virtual machine

1. Choose **Create a resource** in the upper left corner of the Azure portal.
2. In the New page, under Popular, select **Windows Server 2016 Datacenter**.
3. In the Basics tab, under Project details, make sure the correct subscription is selected.
4. For "Resource Group", select **Create new**. Enter *myResourceGroup* as the name and select **Ok**.
5. For **Virtual machine name**, enter *MyVM*.
6. For **Region**, select *(US) East US*.
7. Verify that the **Size** is *Standard D2s v3*.
8. Under **Administrator account**, select **Password**. Enter a user name and a password.

## Create a virtual machine

Complete the Basics tab then Review + create to provision a virtual machine with default customization.

Looking for classic VMs? [Create VM from Azure Marketplace](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups to group your resources.

Subscription \* ⓘ

Free Trial

Resource group \* ⓘ

myResourceGroup

[Create new](#)

### Instance details

Virtual machine name \* ⓘ

myVM

Region \* ⓘ

(US) East US

Availability options ⓘ

No infrastructure redundancy required

Image \* ⓘ

Windows Server 2016 Datacenter

[Browse all public and private images](#)

Size \* ⓘ

**Standard DS1 v2**

1 vcpu, 3.5 GiB memory

[Change size](#)

### Administrator account

Username \* ⓘ

azureUser

Password \* ⓘ

.....

Confirm password \* ⓘ

.....

**Review + create**

< Previous

Next : Disks >

## WARNING

The "Disks" tab features an "Encryption Type" field under Disk options. This field is used to specify encryption options for **Managed Disks + CMK**, not for Azure Disk Encryption.

To avoid confusion, we suggest you skip the *Disks* tab entirely while completing this tutorial.

9. Select the "Management" tab and verify that you have a Diagnostics Storage Account. If you have no storage accounts, select "Create New", give your new account a name, and select "Ok"

The screenshot shows the 'Create a virtual machine' wizard on the 'Management' tab. A red circle highlights the 'Management' tab. A red arrow points from the 'Management' tab to a separate 'Create storage account' dialog box. The 'Create storage account' dialog has the following fields:

- Name: adestorageaccount (with a green checkmark)
- Account kind: Storage (general purpose v1)
- Performance: Standard (selected)
- Replication: Locally-redundant storage (LRS)

The 'Create storage account' dialog also has a note: "No existing storage accounts in current subscription" and a link "Create new".

10. Click "Review + Create".

11. On the **Create a virtual machine** page, you can see the details about the VM you are about to create.  
When you are ready, select **Create**.

It will take a few minutes for your VM to be deployed. When the deployment is finished, move on to the next section.

## Encrypt the virtual machine

1. When the VM deployment is complete, select **Go to resource**.
2. On the left-hand sidebar, select **Disks**.
3. On the top bar, select **Additional Settings**.
4. Under **Encryption settings > Disks to encrypt**, select **OS and data disks**.

## Disk settings

myVM

### Ultra disk

Enable Ultra disk compatibility ⓘ

- Yes  
 No

 Ultra disk is available only for Availability Zones in eastus. [Learn more ↗](#)

### Encryption settings

Azure Disk Encryption (ADE) provides volume encryption for the OS and data disks. [Learn more about Azure Disk Encryption.](#)

#### Disks to encrypt ⓘ

None

None

OS disk

Data disks

OS and data disks



5. Under **Encryption settings**, choose **Select a key vault and key for encryption**.

6. On the **Select key from Azure Key Vault** screen, select **Create New**.

Home > myVM - Disks > Encryption > Select key from Azure Key Vault

### Select key from Azure Key Vault

Key vault \*

Select the key vault.  
[Create new](#)

Key

Select the key.  
Create new

Version ⓘ

Select the version.  
Create new

7. To the left of **Key vault and key**, select **Click to select a key**.

8. On the **Select key from Azure Key Vault**, under the **Key Vault** field, select **Create new**.

9. On the **Create key vault** screen, ensure that the Resource Group is *myResourceGroup*, and give your key vault a name. Every key vault across Azure must have an unique name.

10. On the **Access Policies** tab, check the **Azure Disk Encryption for volume encryption** box.

## Create key vault

Basics

Access policy

Networking

Tags

Review + create

Enable Access to:

Azure Virtual Machines for deployment ⓘ

Azure Resource Manager for template deployment ⓘ

Azure Disk Encryption for volume encryption ⓘ

11. Select **Review + create**.
12. After the key vault has passed validation, select **Create**. This will return you to the **Select key from Azure Key Vault** screen.
13. Leave the **Key** field blank and choose **Select**.
14. At the top of the encryption screen, click **Save**. A popup will warn you that the VM will reboot. Click **Yes**.

## Clean up resources

When no longer needed, you can delete the resource group, virtual machine, and all related resources. To do so, select the resource group for the virtual machine, select **Delete**, then confirm the name of the resource group to delete.

## Next steps

In this quickstart, you created a Key Vault that was enable for encryption keys, created a virtual machine, and enabled the virtual machine for encryption.

[Azure Disk Encryption overview](#)

# Azure Disk Encryption scenarios on Windows VMs

9/21/2022 • 13 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Azure Disk Encryption for Windows virtual machines (VMs) uses the BitLocker feature of Windows to provide full disk encryption of the OS disk and data disk. Additionally, it provides encryption of the temporary disk when the VolumeType parameter is All.

Azure Disk Encryption is [integrated with Azure Key Vault](#) to help you control and manage the disk encryption keys and secrets. For an overview of the service, see [Azure Disk Encryption for Windows VMs](#).

You can only apply disk encryption to virtual machines of [supported VM sizes and operating systems](#). You must also meet the following prerequisites:

- [Networking requirements](#)
- [Group Policy requirements](#)
- [Encryption key storage requirements](#)

## IMPORTANT

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM. See [Azure Disk Encryption with Azure AD \(previous release\)](#) for details.
- You should [take a snapshot](#) and/or create a backup before disks are encrypted. Backups ensure that a recovery option is possible if an unexpected failure occurs during encryption. VMs with managed disks require a backup before encryption occurs. Once a backup is made, you can use the [Set-AzVMDiskEncryptionExtension cmdlet](#) to encrypt managed disks by specifying the -skipVmBackup parameter. For more information about how to back up and restore encrypted VMs, see [Back up and restore encrypted Azure VM](#).
- Encrypting or disabling encryption may cause a VM to reboot.

## Install tools and connect to Azure

Azure Disk Encryption can be enabled and managed through the [Azure CLI](#) and [Azure PowerShell](#). To do so you must install the tools locally and connect to your Azure subscription.

### Azure CLI

The [Azure CLI 2.0](#) is a command-line tool for managing Azure resources. The CLI is designed to flexibly query data, support long-running operations as non-blocking processes, and make scripting easy. You can install it locally by following the steps in [Install the Azure CLI](#).

To [Sign in to your Azure account with the Azure CLI](#), use the `az login` command.

```
az login
```

If you would like to select a tenant to sign in under, use:

```
az login --tenant <tenant>
```

If you have multiple subscriptions and want to specify a specific one, get your subscription list with [az account](#)

list and specify with [az account set](#).

```
az account list
az account set --subscription "<subscription name or ID>"
```

For more information, see [Get started with Azure CLI 2.0](#).

## Azure PowerShell

The [Azure PowerShell az module](#) provides a set of cmdlets that uses the [Azure Resource Manager](#) model for managing your Azure resources. You can use it in your browser with [Azure Cloud Shell](#), or you can install it on your local machine using the instructions in [Install the Azure PowerShell module](#).

If you already have it installed locally, make sure you use the latest version of Azure PowerShell SDK version to configure Azure Disk Encryption. Download the latest version of [Azure PowerShell release](#).

To [Sign in to your Azure account with Azure PowerShell](#), use the [Connect-AzAccount](#) cmdlet.

```
Connect-AzAccount
```

If you have multiple subscriptions and want to specify one, use the [Get-AzSubscription](#) cmdlet to list them, followed by the [Set-AzContext](#) cmdlet:

```
Set-AzContext -Subscription <SubscriptionId>
```

Running the [Get-AzContext](#) cmdlet will verify that the correct subscription has been selected.

To confirm the Azure Disk Encryption cmdlets are installed, use the [Get-command](#) cmdlet:

```
Get-command *diskencryption*
```

For more information, see [Getting started with Azure PowerShell](#).

## Enable encryption on an existing or running Windows VM

In this scenario, you can enable encryption by using the Resource Manager template, PowerShell cmdlets, or CLI commands. If you need schema information for the virtual machine extension, see the [Azure Disk Encryption for Windows extension](#) article.

### Enable encryption on existing or running VMs with Azure PowerShell

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet to enable encryption on a running IaaS virtual machine in Azure.

- **Encrypt a running VM:** The script below initializes your variables and runs the Set-AzVMDiskEncryptionExtension cmdlet. The resource group, VM, and key vault should have already been created as prerequisites. Replace MyKeyVaultResourceGroup, MyVirtualMachineResourceGroup, MySecureVM, and MySecureVault with your values.

```

$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyVaultId $keyVaultResourceId;

```

- Encrypt a running VM using KEK:

```

$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyVaultId $keyVaultResourceId -
KeyEncryptionKeyUrl $keyEncryptionKeyUrl -KeyEncryptionKeyVaultId $keyVaultResourceId;

```

#### **NOTE**

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]

- Verify the disks are encrypted: To check on the encryption status of an IaaS VM, use the [Get-AzVmDiskEncryptionStatus](#) cmdlet.

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

#### **Enable encryption on existing or running VMs with the Azure CLI**

Use the [az vm encryption enable](#) command to enable encryption on a running IaaS virtual machine in Azure.

- Encrypt a running VM:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-encryption-keyvault "MySecureVault" --volume-type [All|OS|Data]
```

- Encrypt a running VM using KEK:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type [All|OS|Data]
```

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

```
/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]
```

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: [https://\[keyvault-name\].vault.azure.net/keys/\[kekname\]/\[kek-unique-id\]](https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id])

- **Verify the disks are encrypted:** To check on the encryption status of an IaaS VM, use the [az vm encryption show](#) command.

```
az vm encryption show --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup"
```

To disable the encryption, see [Disable encryption and remove the encryption extension](#).

#### Using the Resource Manager template

You can enable disk encryption on existing or running IaaS Windows VMs in Azure by using the [Resource Manager template to encrypt a running Windows VM](#).

1. On the Azure quickstart template, click **Deploy to Azure**.
2. Select the subscription, resource group, location, settings, legal terms, and agreement. Click **Purchase** to enable encryption on the existing or running IaaS VM.

The following table lists the Resource Manager template parameters for existing or running VMs:

PARAMETER	DESCRIPTION
vmName	Name of the VM to run the encryption operation.
keyVaultName	Name of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet <pre>(Get-AzKeyVault -ResourceGroupName &lt;MyKeyVaultResourceGroupName&gt;).Vaultname</pre> or the Azure CLI command <pre>az keyvault list --resource-group "MyKeyVaultResourceGroup"</pre>
keyVaultResourceGroup	Name of the resource group that contains the key vault
keyEncryptionKeyURL	The URL of the key encryption key, in the format <a href="https://&lt;keyvault-name&gt;.vault.azure.net/key/&lt;key-name&gt;">https://&lt;keyvault-name&gt;.vault.azure.net/key/&lt;key-name&gt;</a> . If you do not wish to use a KEK, leave this field blank.
volumeType	Type of volume that the encryption operation is performed on. Valid values are <i>OS</i> , <i>Data</i> , and <i>All</i> .
forceUpdateTag	Pass in a unique value like a GUID every time the operation needs to be force run.

PARAMETER	DESCRIPTION
resizeOSDisk	Should the OS partition be resized to occupy full OS VHD before splitting system volume.
location	Location for all resources.

## Enable encryption on NVMe disks for Lsv2 VMs

This scenario describes enabling Azure Disk Encryption on NVMe disks for Lsv2 series VMs. The Lsv2-series features local NVMe storage. Local NVMe Disks are temporary, and data will be lost on these disks if you stop/deallocate your VM (See: [Lsv2-series](#)).

To enable encryption on NVMe disks:

1. Initialize the NVMe disks and create NTFS volumes.
2. Enable encryption on the VM with the VolumeType parameter set to All. This will enable encryption for all OS and data disks, including volumes backed by NVMe disks. For information, see [Enable encryption on an existing or running Windows VM](#).

Encryption will persist on the NVMe disks in the following scenarios:

- VM reboot
- Virtual machine scale set reimage
- Swap OS

NVMe disks will be uninitialized the following scenarios:

- Start VM after deallocation
- Service healing
- Backup

In these scenarios, the NVMe disks need to be initialized after the VM starts. To enable encryption on the NVMe disks, run command to enable Azure Disk Encryption again after the NVMe disks are initialized.

In addition to the scenarios listed in the [Unsupported Scenarios](#) section, encryption of NVMe disks is not supported for:

- VMs encrypted with Azure Disk Encryption with AAD (previous release)
- NVMe disks with storage spaces
- Azure Site Recovery of SKUs with NVMe disks (see [Support matrix for Azure VM disaster recovery between Azure regions: Replicated machines - storage](#)).

## New IaaS VMs created from customer-encrypted VHD and encryption keys

In this scenario, you can create a new VM from a pre-encrypted VHD and the associated encryption keys using PowerShell cmdlets or CLI commands.

Use the instructions in [Prepare a pre-encrypted Windows VHD](#). After the image is created, you can use the steps in the next section to create an encrypted Azure VM.

### Encrypt VMs with pre-encrypted VHDS with Azure PowerShell

You can enable disk encryption on your encrypted VHD by using the PowerShell cmdlet [Set-AzVMOSDisk](#). The example below gives you some common parameters.

```
$VirtualMachine = New-AzVMConfig -VMName "MySecureVM" -VMSize "Standard_A1"
$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name "SecureOSDisk" -VhdUri "os.vhd" Caching ReadWrite
-Windows -CreateOption "Attach" -DiskEncryptionKeyUrl
"https://mytestvault.vault.azure.net/secrets/Test1/514ceb769c984379a7e0230bddaaaaaa" -
DiskEncryptionKeyVaultId "/subscriptions/00000000-0000-0000-
000000000000/resourceGroups/myKvresourcegroup/providers/Microsoft.KeyVault/vaults/mytestvault"
New-AzVM -VM $VirtualMachine -ResourceGroupName "MyVirtualMachineResourceGroup"
```

## Enable encryption on a newly added data disk

You can [add a new disk to a Windows VM using PowerShell](#), or [through the Azure portal](#).

### Enable encryption on a newly added disk with Azure PowerShell

When using PowerShell to encrypt a new disk for Windows VMs, a new sequence version should be specified. The sequence version has to be unique. The script below generates a GUID for the sequence version. In some cases, a newly added data disk might be encrypted automatically by the Azure Disk Encryption extension. Auto encryption usually occurs when the VM reboots after the new disk comes online. This is typically caused because "All" was specified for the volume type when disk encryption previously ran on the VM. If auto encryption occurs on a newly added data disk, we recommend running the Set-AzVmDiskEncryptionExtension cmdlet again with new sequence version. If your new data disk is auto encrypted and you do not wish to be encrypted, decrypt all drives first then re-encrypt with a new sequence version specifying OS for the volume type.

- Encrypt a running VM:** The script below initializes your variables and runs the Set-AzVmDiskEncryptionExtension cmdlet. The resource group, VM, and key vault should have already been created as prerequisites. Replace MyKeyVaultResourceGroup, MyVirtualMachineResourceGroup, MySecureVM, and MySecureVault with your values. This example uses "All" for the -VolumeType parameter, which includes both OS and Data volumes. If you only want to encrypt the OS volume, use "OS" for the -VolumeType parameter.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVmDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyVaultId $keyVaultResourceId -
VolumeType "All" -SequenceVersion $sequenceVersion;
```

- Encrypt a running VM using KEK:** This example uses "All" for the -VolumeType parameter, which includes both OS and Data volumes. If you only want to encrypt the OS volume, use "OS" for the -VolumeType parameter.

```

$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$KeyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $KeyVault.VaultUri;
$keyVaultResourceId = $KeyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;
$sequenceVersion = [Guid]::NewGuid();

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGname -VMName $vmName -
DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -DiskEncryptionKeyId $keyVaultResourceId -
KeyEncryptionKeyUrl $keyEncryptionKeyUrl -KeyEncryptionKeyVaultId $keyVaultResourceId -VolumeType
"All" -SequenceVersion $sequenceVersion;

```

#### **NOTE**

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

```
/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-
name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]
```

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: [https://\[keyvault-name\].vault.azure.net/keys/\[kekname\]/\[kek-unique-id\]](https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id])

### **Enable encryption on a newly added disk with Azure CLI**

The Azure CLI command will automatically provide a new sequence version for you when you run the command to enable encryption. The example uses "All" for the volume-type parameter. You may need to change the volume-type parameter to OS if you're only encrypting the OS disk. In contrast to PowerShell syntax, the CLI does not require the user to provide a unique sequence version when enabling encryption. The CLI automatically generates and uses its own unique sequence version value.

- **Encrypt a running VM:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-
encryption-keyvault "MySecureVault" --volume-type "All"
```

- **Encrypt a running VM using KEK:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --disk-
encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault
"MySecureVaultContainingTheKEK" --volume-type "All"
```

### **Disable encryption and remove the encryption extension**

You can disable the Azure disk encryption extension, and you can remove the Azure disk encryption extension. These are two distinct operations.

To remove ADE, it is recommended that you first disable encryption and then remove the extension. If you remove the encryption extension without disabling it, the disks will still be encrypted. If you disable encryption after removing the extension, the extension will be reinstalled (to perform the decrypt operation) and will need to be removed a second time.

#### **Disable encryption**

You can disable encryption using Azure PowerShell, the Azure CLI, or with a Resource Manager template. Disabling encryption does **not** remove the extension (see [Remove the encryption extension](#)).

#### WARNING

Disabling data disk encryption when both the OS and data disks have been encrypted can have unexpected results. Disable encryption on all disks instead.

Disabling encryption will start a background process of BitLocker to decrypt the disks. This process should be given sufficient time to complete before attempting to any re-enable encryption.

- **Disable disk encryption with Azure PowerShell:** To disable the encryption, use the [Disable-AzVMDiskEncryption](#) cmdlet.

```
Disable-AzVMDiskEncryption -ResourceGroupName "MyVirtualMachineResourceGroup" -VMName "MySecureVM" -VolumeType "all"
```

- **Disable encryption with the Azure CLI:** To disable encryption, use the [az vm encryption disable](#) command.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --volume-type "all"
```

- **Disable encryption with a Resource Manager template:**

1. Click **Deploy to Azure** from the [Disable disk encryption on running Windows VM](#) template.
2. Select the subscription, resource group, location, VM, volume type, legal terms, and agreement.
3. Click **Purchase** to disable disk encryption on a running Windows VM.

#### Remove the encryption extension

If you want to decrypt your disks and remove the encryption extension, you must disable encryption **before** removing the extension; see [disable encryption](#).

You can remove the encryption extension using Azure PowerShell or the Azure CLI.

- **Disable disk encryption with Azure PowerShell:** To remove the encryption, use the [Remove-AzVMDiskEncryptionExtension](#) cmdlet.

```
Remove-AzVMDiskEncryptionExtension -ResourceGroupName "MyVirtualMachineResourceGroup" -VMName "MySecureVM"
```

- **Disable encryption with the Azure CLI:** To remove encryption, use the [az vm extension delete](#) command.

```
az vm extension delete -g "MyVirtualMachineResourceGroup" --vm-name "MySecureVM" -n "AzureDiskEncryptionForWindows"
```

## Unsupported scenarios

Azure Disk Encryption does not work for the following scenarios, features, and technology:

- Encrypting basic tier VM or VMs created through the classic VM creation method.
- Encrypting VMs configured with software-based RAID systems.

- Encrypting VMs configured with Storage Spaces Direct (S2D), or Windows Server versions before 2016 configured with Windows Storage Spaces.
- Integration with an on-premises key management system.
- Azure Files (shared file system).
- Network File System (NFS).
- Dynamic volumes.
- Windows Server containers, which create dynamic volumes for each container.
- Ephemeral OS disks.
- Encryption of shared/distributed file systems like (but not limited to) DFS, GFS, DRDB, and CephFS.
- Moving an encrypted VM to another subscription or region.
- Creating an image or snapshot of an encrypted VM and using it to deploy additional VMs.
- M-series VMs with Write Accelerator disks.
- Applying ADE to a VM that has disks encrypted with [server-side encryption with customer-managed keys](#) (SSE + CMK). Applying SSE + CMK to a data disk on a VM encrypted with ADE is an unsupported scenario as well.
- Migrating a VM that is encrypted with ADE, or has **ever** been encrypted with ADE, to [server-side encryption with customer-managed keys](#).
- Encrypting VMs in failover clusters.
- Encryption of [Azure ultra disks](#).

## Next steps

- [Azure Disk Encryption overview](#)
- [Azure Disk Encryption sample scripts](#)
- [Azure Disk Encryption troubleshooting](#)

# Create and configure a key vault for Azure Disk Encryption on a Windows VM

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Azure Disk Encryption uses Azure Key Vault to control and manage disk encryption keys and secrets. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#).

## WARNING

- If you have previously used Azure Disk Encryption with Azure AD to encrypt a VM, you must continue use this option to encrypt your VM. See [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#) for details.

Creating and configuring a key vault for use with Azure Disk Encryption involves three steps:

## NOTE

You must select the option in the Azure Key Vault access policy settings to enable access to Azure Disk Encryption for volume encryption. If you have enabled the firewall on the key vault, you must go to the Networking tab on the key vault and enable access to Microsoft Trusted Services.

1. Creating a resource group, if needed.
2. Creating a key vault.
3. Setting key vault advanced access policies.

These steps are illustrated in the following quickstarts:

- [Create and encrypt a Windows VM with Azure CLI](#)
- [Create and encrypt a Windows VM with Azure PowerShell](#)

You may also, if you wish, generate or import a key encryption key (KEK).

## NOTE

The steps in this article are automated in the [Azure Disk Encryption prerequisites CLI script](#) and [Azure Disk Encryption prerequisites PowerShell script](#).

## Install tools and connect to Azure

The steps in this article can be completed with the [Azure CLI](#), the [Azure PowerShell Az module](#), or the [Azure portal](#).

While the portal is accessible through your browser, Azure CLI and Azure PowerShell require local installation; see [Azure Disk Encryption for Windows: Install tools](#) for details.

### Connect to your Azure account

Before using the Azure CLI or Azure PowerShell, you must first connect to your Azure subscription. You do so by

[Signing in with Azure CLI](#), [Signing in with Azure PowerShell](#), or supplying your credentials to the Azure portal when prompted.

```
az login
```

```
Connect-AzAccount
```

## Create a resource group

*If you already have a resource group, you can skip to [Create a key vault](#).*

A resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group using the [az group create](#) Azure CLI command, the [New-AzResourceGroup](#) Azure PowerShell command, or from the [Azure portal](#).

- [Azure portal](#)

### Azure CLI

```
az group create --name "myResourceGroup" --location eastus
```

### Azure PowerShell

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

## Create a key vault

*If you already have a key vault, you can skip to [Set key vault advanced access policies](#).*

Create a key vault using the [az keyvault create](#) Azure CLI command, the [New-AzKeyVault](#) Azure PowerShell command, the [Azure portal](#), or a [Resource Manager template](#).

#### WARNING

To ensure that encryption secrets don't cross regional boundaries, you must create and use a key vault that's in the **same region and tenant** as the VMs to be encrypted.

Each Key Vault must have a unique name. Replace <your-unique-keyvault-name> with the name of your key vault in the following examples.

### Azure CLI

When creating a key vault by using the Azure CLI, add the "--enabled-for-disk-encryption" flag.

```
az keyvault create --name "<your-unique-keyvault-name>" --resource-group "myResourceGroup" --location "eastus" --enabled-for-disk-encryption
```

### Azure PowerShell

When creating a key vault using Azure PowerShell, add the "-EnabledForDiskEncryption" flag.

```
New-AzKeyVault -name "<your-unique-keyvault-name>" -ResourceGroupName "myResourceGroup" -Location "eastus" -EnabledForDiskEncryption
```

## Resource Manager template

You can also create a key vault by using the [Resource Manager template](#).

1. On the Azure Quickstart Template, click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, Key Vault name, Object ID, legal terms, and agreement, and then click **Purchase**.

## Set key vault advanced access policies

### IMPORTANT

Newly-created key vaults have soft-delete on by default. If you are using a pre-existing key vault, you **must** enable soft-delete. See [Azure Key Vault soft-delete overview](#).

The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the VM for booting and decrypting the volumes.

If you didn't enable your key vault for disk encryption, deployment, or template deployment at the time of creation (as demonstrated in the previous step), you must update its advanced access policies.

### Azure CLI

Use [az keyvault update](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** Enabled-for-disk-encryption is required.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-disk-encryption "true"
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-deployment "true"
```

- **Enable Key Vault for template deployment, if needed:** Allow Resource Manager to retrieve secrets from the vault.

```
az keyvault update --name "<your-unique-keyvault-name>" --resource-group "MyResourceGroup" --enabled-for-template-deployment "true"
```

### Azure PowerShell

Use the key vault PowerShell cmdlet [Set-AzKeyVaultAccessPolicy](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** EnabledForDiskEncryption is required for Azure Disk encryption.

```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForDiskEncryption
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

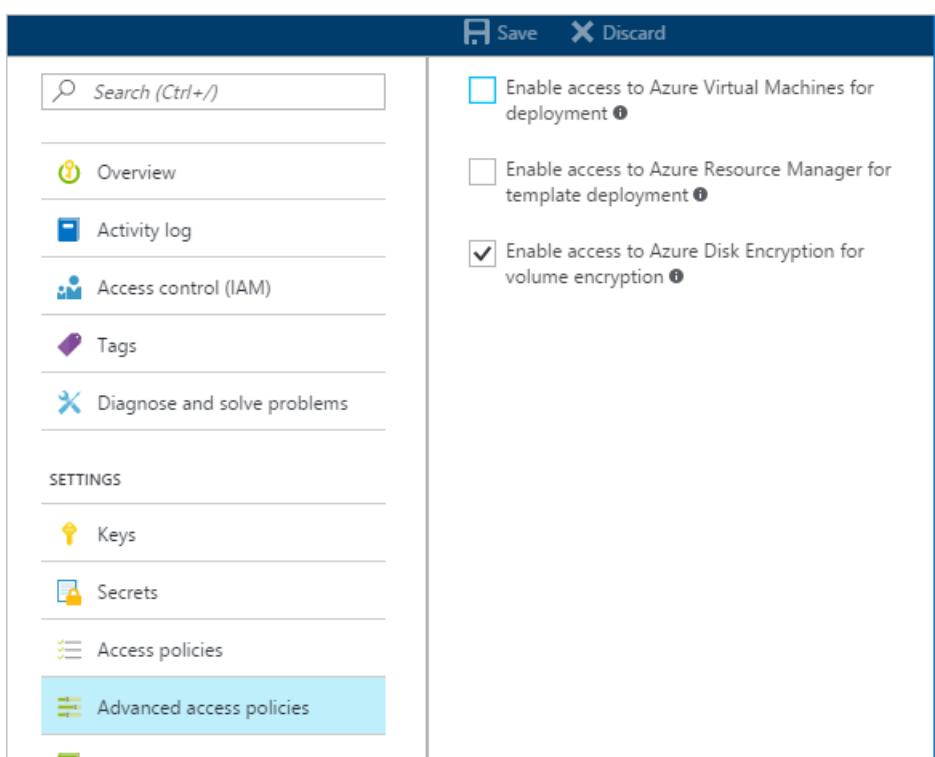
```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForDeployment
```

- **Enable Key Vault for template deployment, if needed:** Enables Azure Resource Manager to get secrets from this key vault when this key vault is referenced in a template deployment.

```
Set-AzKeyVaultAccessPolicy -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "MyResourceGroup" -EnabledForTemplateDeployment
```

## Azure portal

1. Select your key vault and go to **Access Policies**.
2. Under "Enable Access to", select the box labeled **Azure Disk Encryption for volume encryption**.
3. Select **Azure Virtual Machines for deployment** and/or **Azure Resource Manager for template deployment**, if needed.
4. Click **Save**.



## Azure Disk Encryption and auto-rotation

Although Azure Key Vault now has [key auto-rotation](#), it isn't currently compatible with Azure Disk Encryption. Specifically, Azure Disk Encryption will continue to use the original encryption key, even after it has been auto-rotated.

Rotating an encryption key won't break Azure Disk Encryption, but disabling the "old" encryption key (in other

words, the key Azure Disk Encryption is still using) will.

## Set up a key encryption key (KEK)

### IMPORTANT

The account running to enable disk encryption over the key vault must have "reader" permissions.

If you want to use a key encryption key (KEK) for an additional layer of security for encryption keys, add a KEK to your key vault. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault.

You can generate a new KEK by using the Azure CLI `az keyvault key create` command, the Azure PowerShell `Add-AzKeyVaultKey` cmdlet, or the [Azure portal](#). You must generate an RSA key type; Azure Disk Encryption doesn't currently support using Elliptic Curve keys.

You can instead import a KEK from your on-premises key management HSM. For more information, see [Key Vault Documentation](#).

Your key vault KEK URLs must be versioned. Azure enforces this restriction of versioning. For valid secret and KEK URLs, see the following examples:

- Example of a valid secret URL:

<https://contosovault.vault.azure.net/secrets/EncryptionSecretWithKek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

- Example of a valid KEK URL:

<https://contosovault.vault.azure.net/keys/diskencryptionkek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

### Azure CLI

Use the Azure CLI `az keyvault key create` command to generate a new KEK and store it in your key vault.

```
az keyvault key create --name "myKEK" --vault-name "<your-unique-keyvault-name>" --kty RSA --size 4096
```

You may instead import a private key by using the Azure CLI `az keyvault key import` command:

In either case, you supply the name of your KEK to the Azure CLI `az vm encryption enable` `--key-encryption-key` parameter.

```
az vm encryption enable -g "MyResourceGroup" --name "myVM" --disk-encryption-keyvault "<your-unique-keyvault-name>" --key-encryption-key "myKEK"
```

### Azure PowerShell

Use the Azure PowerShell `Add-AzKeyVaultKey` cmdlet to generate a new KEK and store it in your key vault.

```
Add-AzKeyVaultKey -Name "myKEK" -VaultName "<your-unique-keyvault-name>" -Destination "HSM" -Size 4096
```

You may instead import a private key using the Azure PowerShell `az keyvault key import` command.

In either case, you will supply the ID of your KEK key Vault and the URL of your KEK to the Azure PowerShell `Set-AzVMDiskEncryptionExtension` `-KeyEncryptionKeyVaultId` and `-KeyEncryptionKeyUrl` parameters. This example assumes that you are using the same key vault for both the disk encryption key and the KEK.

```
$KeyVault = Get-AzKeyVault -VaultName "<your-unique-keyvault-name>" -ResourceGroupName "myResourceGroup"
$KEK = Get-AzKeyVaultKey -VaultName "<your-unique-keyvault-name>" -Name "myKEK"

Set-AzVMDiskEncryptionExtension -ResourceGroupName MyResourceGroup -VMName "MyVM" -DiskEncryptionKeyVaultUrl
$KeyVault.VaultUri -DiskEncryptionKeyId $KeyVault.ResourceId -KeyEncryptionKeyId $KeyVault.ResourceId
$KeyVault.ResourceId -KeyEncryptionKeyUrl $KEK.Id -SkipVmBackup -VolumeType All
```

## Next steps

- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)
- Learn [Azure Disk Encryption scenarios on Windows VMs](#)
- Learn how to [troubleshoot Azure Disk Encryption](#)
- Read the [Azure Disk Encryption sample scripts](#)

# Azure Disk Encryption sample scripts

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article provides sample scripts for preparing pre-encrypted VHDs and other tasks.

## NOTE

All scripts refer to the latest, non-AAD version of ADE, except where noted.

## Sample PowerShell scripts for Azure Disk Encryption

- **List all encrypted VMs in your subscription**

You can find all ADE-encrypted VMs and the extension version, in all resource groups present in a subscription, using [this PowerShell script](#).

Alternatively, these cmdlets will show all ADE-encrypted VMs (but not the extension version):

```
$osVolEncrypted = {(Get-AzVMDiskEncryptionStatus -ResourceGroupName $_.ResourceGroupName -VMName
$_.Name).OsVolumeEncrypted}
$dataVolEncrypted= {(Get-AzVMDiskEncryptionStatus -ResourceGroupName $_.ResourceGroupName -VMName
$_.Name).DataVolumesEncrypted}
Get-AzVm | Format-Table @{Label="MachineName"; Expression={$.Name}}, @{Label="OsVolumeEncrypted";
Expression=$osVolEncrypted}, @{Label="DataVolumesEncrypted"; Expression=$dataVolEncrypted}
```

- **List all encrypted VMSS instances in your subscription**

You can find all ADE-encrypted VMSS instances and the extension version, in all resource groups present in a subscription, using [this PowerShell script](#).

- **List all disk encryption secrets used for encrypting VMs in a key vault**

```
Get-AzKeyVaultSecret -VaultName $KeyVaultName | where {$_.Tags.ContainsKey('DiskEncryptionKeyFileName')} |
format-table @{Label="MachineName"; Expression={$_.Tags['MachineName']}}, @{Label="VolumeLetter";
Expression={$_.Tags['VolumeLetter']}}, @{Label="EncryptionKeyURL"; Expression={$_.Id}}
```

## Using the Azure Disk Encryption prerequisites PowerShell script

If you're already familiar with the prerequisites for Azure Disk Encryption, you can use the [Azure Disk Encryption prerequisites PowerShell script](#). For an example of using this PowerShell script, see the [Encrypt a VM Quickstart](#). You can remove the comments from a section of the script, starting at line 211, to encrypt all disks for existing VMs in an existing resource group.

The following table shows which parameters can be used in the PowerShell script:

PARAMETER	DESCRIPTION	MANDATORY?
\$resourceGroupName	Name of the resource group to which the KeyVault belongs to. A new resource group with this name will be created if one doesn't exist.	True

PARAMETER	DESCRIPTION	MANDATORY?
\$keyVaultName	Name of the KeyVault in which encryption keys are to be placed. A new vault with this name will be created if one doesn't exist.	True
\$location	Location of the KeyVault. Make sure the KeyVault and VMs to be encrypted are in the same location. Get a location list with <code>Get-AzLocation</code> .	True
\$subscriptionId	Identifier of the Azure subscription to be used. You can get your Subscription ID with <code>Get-AzSubscription</code> .	True
\$aadAppName	Name of the Azure AD application that will be used to write secrets to KeyVault. A new application with this name will be created if one doesn't exist. If this app already exists, pass <code>aadClientSecret</code> parameter to the script.	False
\$aadClientSecret	Client secret of the Azure AD application that was created earlier.	False
\$keyEncryptionKeyName	Name of optional key encryption key in KeyVault. A new key with this name will be created if one doesn't exist.	False

## Resource Manager templates

### Encrypt or decrypt VMs without an Azure AD app

- [Enable disk encryption on an existing or running Windows VM](#)
- [Disable encryption on a running Windows VM](#)

### Encrypt or decrypt VMs with an Azure AD app (previous release)

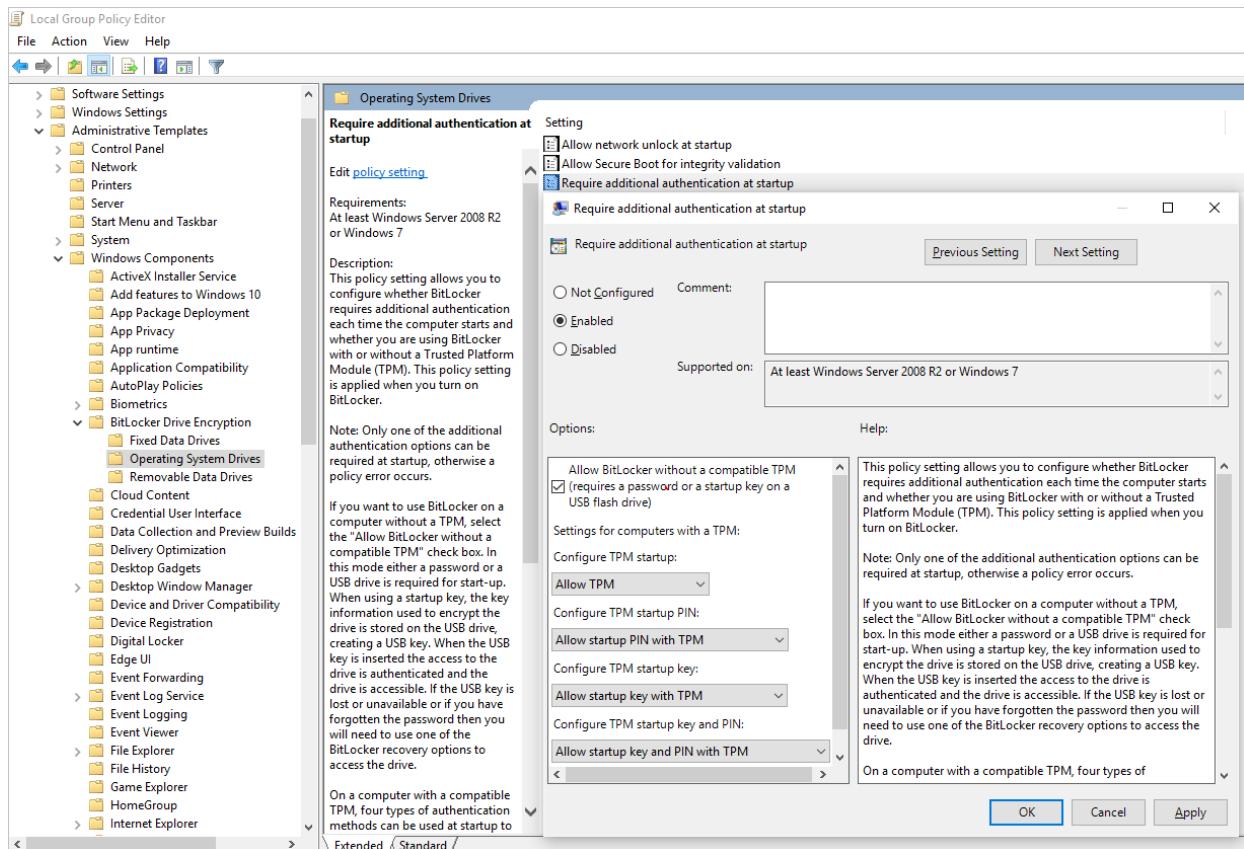
- [Enable disk encryption on an existing or running Windows VM](#)
- [Disable encryption on a running Windows VM](#)
- [Create a new encrypted managed disk from a pre-encrypted VHD/storage blob](#)
  - Creates a new encrypted managed disk provided a pre-encrypted VHD and its corresponding encryption settings

## Prepare a pre-encrypted Windows VHD

The sections that follow are necessary to prepare a pre-encrypted Windows VHD for deployment as an encrypted VHD in Azure IaaS. Use the information to prepare and boot a fresh Windows VM (VHD) on Azure Site Recovery or Azure. For more information on how to prepare and upload a VHD, see [Upload a generalized VHD and use it to create new VMs in Azure](#).

## Update group policy to allow non-TPM for OS protection

Configure the BitLocker Group Policy setting **BitLocker Drive Encryption**, which you'll find under **Local Computer Policy > Computer Configuration > Administrative Templates > Windows Components**. Change this setting to **Operating System Drives > Require additional authentication at startup > Allow BitLocker without a compatible TPM**, as shown in the following figure:



## Install BitLocker feature components

For Windows Server 2012 and later, use the following command:

```
dism /online /Enable-Feature /all /FeatureName:BitLocker /quiet /norestart
```

For Windows Server 2008 R2, use the following command:

```
ServerManagerCmd -install BitLockers
```

## Prepare the OS volume for BitLocker by using `bdehdcfg`

To compress the OS partition and prepare the machine for BitLocker, execute the `bdehdcfg` if needed:

```
bdehdcfg -target c: shrink -quiet
```

## Protect the OS volume by using BitLocker

Use the `manage-bde` command to enable encryption on the boot volume using an external key protector. Also place the external key (.bek file) on the external drive or volume. Encryption is enabled on the system/boot volume after the next reboot.

```
manage-bde -on %systemdrive% -sk [ExternalDriveOrVolume]  
reboot
```

#### NOTE

Prepare the VM with a separate data/resource VHD for getting the external key by using BitLocker.

## Upload encrypted VHD to an Azure storage account

After BitLocker encryption is enabled, the local encrypted VHD needs to be uploaded to your storage account.

```
Add-AzVhd [-Destination] <Uri> [-LocalFilePath] <FileInfo> [[-NumberOfUploaderThreads] <Int32> ] [[-BaseImageUriToPatch] <Uri> ] [[-OverWrite]] [ <CommonParameters>]
```

## Upload the secret for the pre-encrypted VM to your key vault

The disk encryption secret that you obtained previously must be uploaded as a secret in your key vault. This requires granting the set secret permission and the wrapkey permission to the account that will upload the secrets.

```
# Typically, account Id is the user principal name (in user@domain.com format)
$upn = (Get-AzureRmContext).Account.Id
Set-AzKeyVaultAccessPolicy -VaultName $kvname -UserPrincipalName $acctid -PermissionsToKeys wrapKey -
PermissionsToSecrets set

# In cloud shell, the account ID is a managed service identity, so specify the username directly
# $upn = "user@domain.com"
# Set-AzKeyVaultAccessPolicy -VaultName $kvname -UserPrincipalName $acctid -PermissionsToKeys wrapKey -
PermissionsToSecrets set

# When running as a service principal, retrieve the service principal ID from the account ID, and set access
policy to that
# $acctid = (Get-AzureRmContext).Account.Id
# $spoid = (Get-AzureRmADServicePrincipal -ServicePrincipalName $acctid).Id
# Set-AzKeyVaultAccessPolicy -VaultName $kvname -ObjectId $spoid -BypassObjectIdValidation -
PermissionsToKeys wrapKey -PermissionsToSecrets set
```

### Disk encryption secret not encrypted with a KEK

To set up the secret in your key vault, use [Set-AzKeyVaultSecret](#). The passphrase is encoded as a base64 string and then uploaded to the key vault. In addition, make sure that the following tags are set when you create the secret in the key vault.

```
# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

$tags = @{
    "DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP";
    "DiskEncryptionKeyFileName" =
    "LinuxPassPhraseFileName"
}
$secretName = [guid]::NewGuid().ToString()
$secretValue = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($passphrase))
$secureSecretValue = ConvertTo-SecureString $secretValue -AsPlainText -Force

$secret = Set-AzKeyVaultSecret -VaultName $KeyVaultName -Name $secretName -SecretValue $secureSecretValue -
tags $tags
$secretUrl = $secret.Id
```

Use the `$secretUrl` in the next step for [attaching the OS disk without using KEK](#).

### Disk encryption secret encrypted with a KEK

Before you upload the secret to the key vault, you can optionally encrypt it by using a key encryption key. Use the wrap API to first encrypt the secret using the key encryption key. The output of this wrap operation is a base64 URL encoded string, which you can then upload as a secret by using the [Set-AzKeyVaultSecret](#) cmdlet.

```
# This is the passphrase that was provided for encryption during the distribution installation
$passphrase = "contoso-password"

Add-AzKeyVaultKey -VaultName $KeyVaultName -Name "keyencryptionkey" -Destination Software
$keyEncryptionKey = Get-AzKeyVaultKey -VaultName $KeyVault.OriginalVault.Name -Name "keyencryptionkey"

$apiversion = "2015-06-01"

#####
# Get Auth URI
#####

$uri = $KeyVault.VaultUri + "/keys"
$headers = @{}

$response = try { Invoke-RestMethod -Method GET -Uri $uri -Headers $headers } catch {
$_.Exception.Response }

$authHeader = $response.Headers["www-authenticate"]
$authUri = [regex]::match($authHeader, 'authorization="(.*)"').Groups[1].Value

Write-Host "Got Auth URI successfully"

#####
# Get Auth Token
#####

$uri = $authUri + "/oauth2/token"
$body = "grant_type=client_credentials"
$body += "&client_id=" + $AadClientId
$body += "&client_secret=" + [Uri]::EscapeDataString($AadClientSecret)
$body += "&resource=" + [Uri]::EscapeDataString("https://vault.azure.net")
$headers = @{}

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$access_token = $response.access_token

Write-Host "Got Auth Token successfully"

#####
# Get KEK info
#####

$uri = $KeyEncryptionKey.Id + "?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token}

$response = Invoke-RestMethod -Method GET -Uri $uri -Headers $headers

$keyid = $response.key.kid

Write-Host "Got KEK info successfully"

#####
# Encrypt passphrase using KEK
#####

$passphraseB64 = [Convert]::ToBase64String([System.Text.Encoding]::ASCII.GetBytes($Passphrase))
$uri = $keyid + "/encrypt?api-version=" + $apiversion
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"alg" = "RSA-OAEP"; "value" = $passphraseB64}
$body = $bodyObj | ConvertTo-Json
```

```

$response = Invoke-RestMethod -Method POST -Uri $uri -Headers $headers -Body $body

$wrappedSecret = $response.value

Write-Host "Encrypted passphrase successfully"

#####
# Store secret
#####

$secretName = [guid]::NewGuid().ToString()
$uri = $KeyVault.VaultUri + "/secrets/" + $secretName + "?api-version=" + $apiversion
$secretAttributes = @{"enabled" = $true}
$secretTags = @{"DiskEncryptionKeyEncryptionAlgorithm" = "RSA-OAEP"; "DiskEncryptionKeyFileName" =
"LinuxPassPhraseFileName"}
$headers = @{"Authorization" = "Bearer " + $access_token; "Content-Type" = "application/json"}
$bodyObj = @{"value" = $wrappedSecret; "attributes" = $secretAttributes; "tags" = $secretTags}
$body = $bodyObj | ConvertTo-Json

$response = Invoke-RestMethod -Method PUT -Uri $uri -Headers $headers -Body $body

Write-Host "Stored secret successfully"

$secretUrl = $response.id

```

Use `$KeyEncryptionKey` and `$secretUrl` in the next step for [attaching the OS disk using KEK](#).

## Specify a secret URL when you attach an OS disk

### Without using a KEK

While you're attaching the OS disk, you need to pass `$secretUrl`. The URL was generated in the "Disk-encryption secret not encrypted with a KEK" section.

```

Set-AzVMOSDisk ` 
    -VM $VirtualMachine ` 
    -Name $OSDiskName ` 
    -SourceImageUri $VhdUri ` 
    -VhdUri $OSDiskUri ` 
    -Windows ` 
    -CreateOption FromImage ` 
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -DiskEncryptionKeyUrl $SecretUrl

```

### Using a KEK

When you attach the OS disk, pass `$KeyEncryptionKey` and `$secretUrl`. The URL was generated in the "Disk encryption secret encrypted with a KEK" section.

```

Set-AzVMOSDisk ` 
    -VM $VirtualMachine ` 
    -Name $OSDiskName ` 
    -SourceImageUri $CopiedTemplateBlobUri ` 
    -VhdUri $OSDiskUri ` 
    -Windows ` 
    -CreateOption FromImage ` 
    -DiskEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -DiskEncryptionKeyUrl $SecretUrl ` 
    -KeyEncryptionKeyVaultId $KeyVault.ResourceId ` 
    -KeyEncryptionKeyURL $KeyEncryptionKey.Id

```

# Azure Disk Encryption troubleshooting guide

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

This guide is for IT professionals, information security analysts, and cloud administrators whose organizations use Azure Disk Encryption. This article is to help with troubleshooting disk-encryption-related problems.

Before taking any of the steps below, first ensure that the VMs you are attempting to encrypt are among the [supported VM sizes and operating systems](#), and that you have met all the prerequisites:

- [Networking requirements](#)
- [Group policy requirements](#)
- [Encryption key storage requirements](#)

## Troubleshooting 'Failed to send DiskEncryptionData'

When encrypting a VM fails with the error message "Failed to send DiskEncryptionData...", it is usually caused by one of the following situations:

- Having the Key Vault existing in a different region and/or subscription than the Virtual Machine
- Advanced access policies in the Key Vault are not set to allow Azure Disk Encryption
- Key Encryption Key, when in use, has been disabled or deleted in the Key Vault
- Typo in the Resource ID or URL for the Key Vault or Key Encryption Key (KEK)
- Special characters used while naming the VM, data disks, or keys. i.e \_VMName, élite, etc
- Unsupported encryption scenarios
- Network issues that prevent the VM/Host from accessing the required resources

### Suggestions

- Make sure the Key Vault exists in the same region and subscription as the Virtual Machine
- Ensure that you have [set key vault advanced access policies](#) properly
- If you are using KEK, ensure the key exists and is enabled in Key Vault
- Check VM name, data disks, and keys follow [key vault resource naming restrictions](#)
- Check for any typos in the Key Vault name or KEK name in your PowerShell or CLI command

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string: /subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]

- Ensure you are not following any [unsupported scenario](#)
- Ensure you are meeting [network requirements](#) and try again

## Troubleshooting Azure Disk Encryption behind a firewall

When connectivity is restricted by a firewall, proxy requirement, or network security group (NSG) settings, the ability of the extension to perform needed tasks might be disrupted. This disruption can result in status

messages such as "Extension status not available on the VM." In expected scenarios, the encryption fails to finish. The sections that follow have some common firewall problems that you might investigate.

## Network security groups

Any network security group settings that are applied must still allow the endpoint to meet the documented network configuration [prerequisites](#) for disk encryption.

## Azure Key Vault behind a firewall

When encryption is being enabled with [Azure AD credentials](#), the target VM must allow connectivity to both Azure Active Directory endpoints and Key Vault endpoints. Current Azure Active Directory authentication endpoints are maintained in sections 56 and 59 of the [Microsoft 365 URLs and IP address ranges](#) documentation. Key Vault instructions are provided in the documentation on how to [Access Azure Key Vault behind a firewall](#).

## Azure Instance Metadata Service

The VM must be able to access the [Azure Instance Metadata service](#) endpoint ( `169.254.169.254` ) and the [virtual public IP address](#) ( `168.63.129.16` ) used for communication with Azure platform resources. Proxy configurations that alter local HTTP traffic to these addresses (for example, adding an X-Forwarded-For header) are not supported.

# Troubleshooting Windows Server 2016 Server Core

On Windows Server 2016 Server Core, the bdehdcfg component isn't available by default. This component is required by Azure Disk Encryption. It's used to split the system volume from OS volume, which is done only once for the life time of the VM. These binaries aren't required during later encryption operations.

To work around this issue, copy the following four files from a Windows Server 2016 Data Center VM to the same location on Server Core:

```
\windows\system32\bdehdcfg.exe  
\windows\system32\bdehdcfglib.dll  
\windows\system32\en-US\bdehdcfglib.dll.mui  
\windows\system32\en-US\bdehdcfg.exe.mui
```

1. Enter the following command:

```
bdehdcfg.exe -target default
```

2. This command creates a 550-MB system partition. Reboot the system.

3. Use DiskPart to check the volumes, and then proceed.

For example:

```
DISKPART> list vol

Volume ### Ltr Label Fs Type Size Status Info
----- -- -----
Volume 0 C NTFS Partition 126 GB Healthy Boot
Volume 1 NTFS Partition 550 MB Healthy System
Volume 2 D Temporary S NTFS Partition 13 GB Healthy Pagefile
```

## Troubleshooting encryption status

The portal may display a disk as encrypted even after it has been unencrypted within the VM. This can occur

when low-level commands are used to directly unencrypt the disk from within the VM, instead of using the higher level Azure Disk Encryption management commands. The higher level commands not only unencrypt the disk from within the VM, but outside of the VM they also update important platform level encryption settings and extension settings associated with the VM. If these are not kept in alignment, the platform will not be able to report encryption status or provision the VM properly.

To disable Azure Disk Encryption with PowerShell, use [Disable-AzVMDiskEncryption](#) followed by [Remove-AzVMDiskEncryptionExtension](#). Running Remove-AzVMDiskEncryptionExtension before the encryption is disabled will fail.

To disable Azure Disk Encryption with CLI, use `az vm encryption disable`.

## Next steps

In this document, you learned more about some common problems in Azure Disk Encryption and how to troubleshoot those problems. For more information about this service and its capabilities, see the following articles:

- [Apply disk encryption in Microsoft Defender for Cloud](#)
- [Azure data encryption at rest](#)



# Azure Disk Encryption with Azure AD (previous release)

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure AD application parameter to enable VM disk encryption. With the new release, you are no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters using the new release. To view instructions to enable VM disk encryption using the new release, see [Azure Disk Encryption for Windows VMs](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

This article supplements [Azure Disk Encryption for Windows VMs](#) with additional requirements and prerequisites for Azure Disk Encryption with Azure AD (previous release). The [Supported VMs and operating systems](#) section remains the same.

## Networking and Group Policy

To enable the Azure Disk Encryption feature using the older AAD parameter syntax, the IaaS VMs must meet the following network endpoint configuration requirements:

- To get a token to connect to your key vault, the IaaS VM must be able to connect to an Azure Active Directory endpoint, [login.microsoftonline.com].
- To write the encryption keys to your key vault, the IaaS VM must be able to connect to the key vault endpoint.
- The IaaS VM must be able to connect to an Azure storage endpoint that hosts the Azure extension repository and an Azure storage account that hosts the VHD files.
- If your security policy limits access from Azure VMs to the Internet, you can resolve the preceding URI and configure a specific rule to allow outbound connectivity to the IPs. For more information, see [Azure Key Vault behind a firewall](#).
- The VM to be encrypted must be configured to use TLS 1.2 as the default protocol. If TLS 1.0 has been explicitly disabled and the .NET version has not been updated to 4.6 or higher, the following registry change will enable ADE to select the more recent TLS version:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\.NETFramework\v4.0.30319]
"SystemDefaultTlsVersions"=dword:00000001
"SchUseStrongCrypto"=dword:00000001

[HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\.NETFramework\v4.0.30319]
"SystemDefaultTlsVersions"=dword:00000001
"SchUseStrongCrypto"=dword:00000001`
```

### Group Policy:

- The Azure Disk Encryption solution uses the BitLocker external key protector for Windows IaaS VMs. For domain joined VMs, don't push any group policies that enforce TPM protectors. For information about the group policy for "Allow BitLocker without a compatible TPM," see [BitLocker Group Policy Reference](#).
- BitLocker policy on domain joined virtual machines with custom group policy must include the following

setting: [Configure user storage of BitLocker recovery information -> Allow 256-bit recovery key](#). Azure Disk Encryption will fail when custom group policy settings for BitLocker are incompatible. On machines that didn't have the correct policy setting, apply the new policy, force the new policy to update (gpupdate.exe /force), and then restarting may be required.

## Encryption key storage requirements

Azure Disk Encryption requires an Azure Key Vault to control and manage disk encryption keys and secrets. Your key vault and VMs must reside in the same Azure region and subscription.

For details, see [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#).

## Next steps

- [Creating and configuring a key vault for Azure Disk Encryption with Azure AD \(previous release\)](#)
- [Enable Azure Disk Encryption with Azure AD on Windows VMs \(previous release\)](#)
- [Azure Disk Encryption prerequisites CLI script](#)
- [Azure Disk Encryption prerequisites PowerShell script](#)

# Creating and configuring a key vault for Azure Disk Encryption with Azure AD (previous release)

9/21/2022 • 15 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure AD application parameter to enable VM disk encryption. With the new release, you are no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters using the new release. To view instructions to enable VM disk encryption using the new release, see [Azure Disk Encryption](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

Azure Disk Encryption uses Azure Key Vault to control and manage disk encryption keys and secrets. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#).

Creating and configuring a key vault for use with Azure Disk Encryption with Azure AD (previous release) involves three steps:

1. Create a key vault.
2. Set up an Azure AD application and service principal.
3. Set the key vault access policy for the Azure AD app.
4. Set key vault advanced access policies.

You may also, if you wish, generate or import a key encryption key (KEK).

See the main [Creating and configuring a key vault for Azure Disk Encryption](#) article for steps on how to [Install tools and connect to Azure](#).

## NOTE

The steps in this article are automated in the [Azure Disk Encryption prerequisites CLI script](#) and [Azure Disk Encryption prerequisites PowerShell script](#).

## Create a key vault

Azure Disk Encryption is integrated with [Azure Key Vault](#) to help you control and manage the disk-encryption keys and secrets in your key vault subscription. You can create a key vault or use an existing one for Azure Disk Encryption. For more information about key vaults, see [Get started with Azure Key Vault](#) and [Secure your key vault](#). You can use a Resource Manager template, Azure PowerShell, or the Azure CLI to create a key vault.

## WARNING

In order to make sure the encryption secrets don't cross regional boundaries, Azure Disk Encryption needs the Key Vault and the VMs to be co-located in the same region. Create and use a Key Vault that is in the same region as the VM to be encrypted.

## Create a key vault with PowerShell

You can create a key vault with Azure PowerShell using the [New-AzKeyVault](#) cmdlet. For additional cmdlets for Key Vault, see [Az.KeyVault](#).

1. Create a new resource group, if needed, with [New-AzResourceGroup](#). To list data center locations, use [Get-AzLocation](#).

```
# Get-AzLocation  
New-AzResourceGroup -Name 'MyKeyVaultResourceGroup' -Location 'East US'
```

2. Create a new key vault using [New-AzKeyVault](#)

```
New-AzKeyVault -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -Location 'East US'
```

3. Note the **Vault Name**, **Resource Group Name**, **Resource ID**, **Vault URI**, and the **Object ID** that are returned for later use when you encrypt the disks.

### Create a key vault with Azure CLI

You can manage your key vault with Azure CLI using the [az keyvault](#) commands. To create a key vault, use [az keyvault create](#).

1. Create a new resource group, if needed, with [az group create](#). To list locations, use [az account list-locations](#)

```
# To list locations: az account list-locations --output table  
az group create -n "MyKeyVaultResourceGroup" -l "East US"
```

2. Create a new key vault using [az keyvault create](#).

```
az keyvault create --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --location "East US"
```

3. Note the **Vault Name** (name), **Resource Group Name**, **Resource ID** (ID), **Vault URI**, and the **Object ID** that are returned for use later.

### Create a key vault with a Resource Manager template

You can create a key vault by using the [Resource Manager template](#).

1. On the Azure quickstart template, click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, Key Vault name, Object ID, legal terms, and agreement, and then click **Purchase**.

## Set up an Azure AD app and service principal

When you need encryption to be enabled on a running VM in Azure, Azure Disk Encryption generates and writes the encryption keys to your key vault. Managing encryption keys in your key vault requires Azure AD authentication. Create an Azure AD application for this purpose. For authentication purposes, you can use either client secret-based authentication or [client certificate-based Azure AD authentication](#).

### Set up an Azure AD app and service principal with Azure PowerShell

To execute the following commands, get and use the [Azure AD PowerShell module](#).

1. Use the [New-AzADApplication](#) PowerShell cmdlet to create an Azure AD application. MyApplicationHomePage and the MyApplicationUri can be any values you wish.

```

$aadClientSecret = "My AAD client secret"
$aadClientSecretSec = ConvertTo-SecureString -String $aadClientSecret -AsPlainText -Force
$azureAdApplication = New-AzADApplication -DisplayName "My Application Display Name" -HomePage
"https://MyApplicationHomePage" -IdentifierUris "https://MyApplicationUri" -Password
$aadClientSecretSec
$servicePrincipal = New-AzADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId -Role
Contributor

```

- The \$azureAdApplication.ApplicationId is the Azure AD ClientID and the \$aadClientSecret is the client secret that you will use later to enable Azure Disk Encryption. Safeguard the Azure AD client secret appropriately. Running `$azureAdApplication.ApplicationId` will show you the ApplicationID.

## Set up an Azure AD app and service principal with Azure CLI

You can manage your service principals with Azure CLI using the [az ad sp](#) commands. For more information, see [Create an Azure service principal](#).

- Create a new service principal.

```

az ad sp create-for-rbac --name "ServicePrincipalName" --password "My-AAD-client-secret" --role
Contributor --scopes /subscriptions/<subscription_id>

```

- The appId returned is the Azure AD ClientID used in other commands. It's also the SPN you'll use for az keyvault set-policy. The password is the client secret that you should use later to enable Azure Disk Encryption. Safeguard the Azure AD client secret appropriately.

## Set up an Azure AD app and service principal through the Azure portal

Use the steps from the [Use portal to create an Azure Active Directory application and service principal that can access resources](#) article to create an Azure AD application. Each step listed below will take you directly to the article section to complete.

- [Verify required permissions](#)
- [Create an Azure Active Directory application](#)
  - You can use any name and sign-on URL you would like when creating the application.
- [Get the application ID and the authentication key](#).
  - The authentication key is the client secret and is used as the AadClientSecret for Set-AzVMDiskEncryptionExtension.
    - The authentication key is used by the application as a credential to sign in to Azure AD. In the Azure portal, this secret is called keys, but has no relation to key vaults. Secure this secret appropriately.
  - The application ID will be used later as the AadClientId for Set-AzVMDiskEncryptionExtension and as the ServicePrincipalName for Set-AzKeyVaultAccessPolicy.

## Set the key vault access policy for the Azure AD app

To write encryption secrets to a specified Key Vault, Azure Disk Encryption needs the Client ID and the Client Secret of the Azure Active Directory application that has permissions to write secrets to the Key Vault.

### NOTE

Azure Disk Encryption requires you to configure the following access policies to your Azure AD client application: *WrapKey* and *Set* permissions.

## Set the key vault access policy for the Azure AD app with Azure PowerShell

Your Azure AD application needs rights to access the keys or secrets in the vault. Use the [Set-AzKeyVaultAccessPolicy](#) cmdlet to grant permissions to the application, using the client ID (which was generated when the application was registered) as the *-ServicePrincipalName* parameter value. To learn more, see the blog post [Azure Key Vault - Step by Step](#).

1. Set the key vault access policy for the AD application with PowerShell.

```
$keyVaultName = 'MySecureVault'  
$aadClientID = 'MyAadAppClientID'  
$KVRGname = 'MyKeyVaultResourceGroup'  
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -  
PermissionsToKeys 'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname
```

### Set the key vault access policy for the Azure AD app with Azure CLI

Use [az keyvault set-policy](#) to set the access policy. For more information, see [Manage Key Vault using CLI 2.0](#).

Give the service principal you created via the Azure CLI access to get secrets and wrap keys with the following command:

```
az keyvault set-policy --name "MySecureVault" --spn "<spn created with CLI/the Azure AD ClientID>" --key-  
permissions wrapKey --secret-permissions set
```

### Set the key vault access policy for the Azure AD app with the portal

1. Open the resource group with your key vault.
2. Select your key vault, go to **Access Policies**, then click **Add new**.
3. Under **Select principal**, search for the Azure AD application you created and select it.
4. For **Key permissions**, check **Wrap Key** under **Cryptographic Operations**.
5. For **Secret permissions**, check **Set** under **Secret Management Operations**.
6. Click **OK** to save the access policy.

Add new permissions    -    □    X

Add a new access policy - PREVIEW

\* Select principal  
vmencrypt >

Configure from template (optional)

Key permissions  
1 selected >

Secret permissions  
1 selected >

Authorized application ⓘ  
None selected

Key permissions

All Key Operations

All

Key Management Operations

Get

List

Update

Create

Import

Delete

Backup

Restore

Cryptographic Operations

Decrypt

Encrypt

UnwrapKey

WrapKey

Verify

Sign

Add new permissions    -    □    X

Add a new access policy - PREVIEW

\* Select principal  
vmencrypt >

Configure from template (optional)

Key permissions  
1 selected >

Secret permissions  
1 selected >

Authorized application ⓘ  
None selected

Secret permissions

All Secret Operations

All

Secret Management Operations

Get

List

Set

Delete

# Set key vault advanced access policies

The Azure platform needs access to the encryption keys or secrets in your key vault to make them available to the VM for booting and decrypting the volumes. Enable disk encryption on the key vault or deployments will fail.

## Set key vault advanced access policies with Azure PowerShell

Use the key vault PowerShell cmdlet [Set-AzKeyVaultAccessPolicy](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** EnabledForDiskEncryption is required for Azure Disk encryption.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForDiskEncryption
```

- **Enable Key Vault for deployment, if needed:** Enables the Microsoft.Compute resource provider to retrieve secrets from this key vault when this key vault is referenced in resource creation, for example when creating a virtual machine.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForDeployment
```

- **Enable Key Vault for template deployment, if needed:** Enables Azure Resource Manager to get secrets from this key vault when this key vault is referenced in a template deployment.

```
Set-AzKeyVaultAccessPolicy -VaultName 'MySecureVault' -ResourceGroupName 'MyKeyVaultResourceGroup' -  
EnabledForTemplateDeployment
```

## Set key vault advanced access policies using the Azure CLI

Use [az keyvault update](#) to enable disk encryption for the key vault.

- **Enable Key Vault for disk encryption:** Enabled-for-disk-encryption is required.

```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
disk-encryption "true"
```

- **Enable Key Vault for deployment, if needed:** Allow Virtual Machines to retrieve certificates stored as secrets from the vault.

```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
deployment "true"
```

- **Enable Key Vault for template deployment, if needed:** Allow Resource Manager to retrieve secrets from the vault.

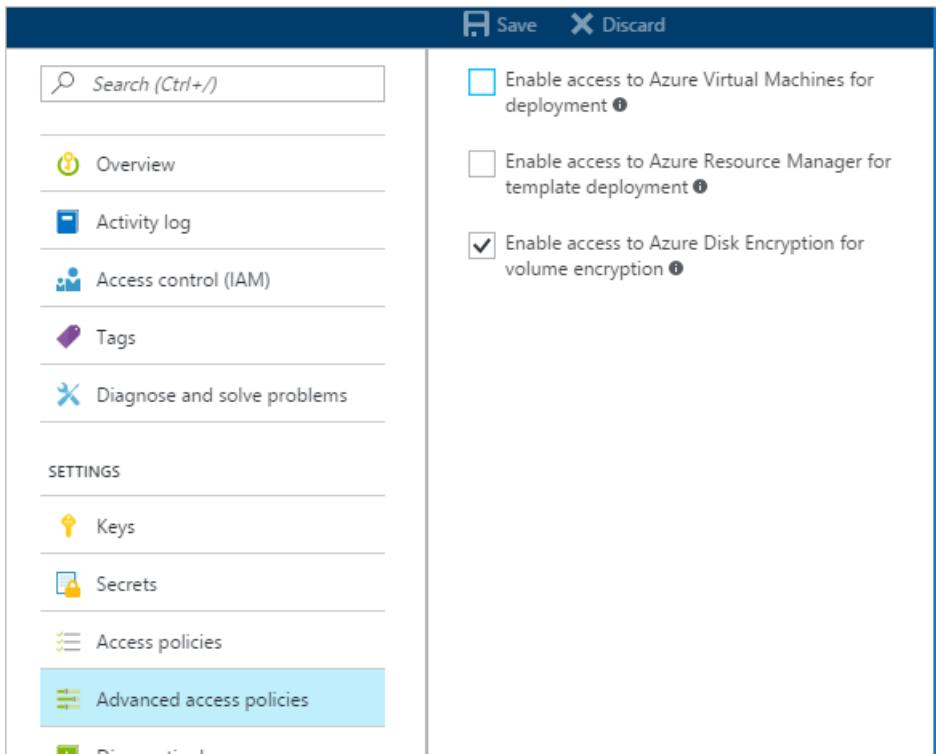
```
az keyvault update --name "MySecureVault" --resource-group "MyKeyVaultResourceGroup" --enabled-for-  
template-deployment "true"
```

## Set key vault advanced access policies through the Azure portal

1. Select your keyvault, go to Access Policies, and Click to show advanced access policies.
2. Select the box labeled Enable access to Azure Disk Encryption for volume encryption.

3. Select **Enable access to Azure Virtual Machines for deployment** and/or **Enable Access to Azure Resource Manager for template deployment**, if needed.

4. Click **Save**.



## Set up a key encryption key (optional)

If you want to use a key encryption key (KEK) for an additional layer of security for encryption keys, add a KEK to your key vault. Use the [Add-AzKeyVaultKey](#) cmdlet to create a key encryption key in the key vault. You can also import a KEK from your on-premises key management HSM. For more information, see [Key Vault Documentation](#). When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault.

- When generating keys, use an RSA key type. Azure Disk Encryption does not yet support using Elliptic Curve keys.
- Your key vault secret and KEK URLs must be versioned. Azure enforces this restriction of versioning. For valid secret and KEK URLs, see the following examples:
  - Example of a valid secret URL:  
<https://contosovault.vault.azure.net/secrets/EncryptionSecretWithKek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - Example of a valid KEK URL:  
<https://contosovault.vault.azure.net/keys/diskencryptionkek/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
- Azure Disk Encryption doesn't support specifying port numbers as part of key vault secrets and KEK URLs. For examples of non-supported and supported key vault URLs, see the following examples:
  - Unacceptable key vault URL  
<https://contosovault.vault.azure.net:443/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>
  - Acceptable key vault URL:  
<https://contosovault.vault.azure.net/secrets/contososecret/xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

### Set up a key encryption key with Azure PowerShell

Before using the PowerShell script, you should be familiar with the Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment. This script creates all Azure Disk Encryption prerequisites and encrypts an existing IaaS VM, wrapping the disk encryption

key by using a key encryption key.

```
# Step 1: Create a new resource group and key vault in the same location.
# Fill in 'MyLocation', 'MyKeyVaultResourceGroup', and 'MySecureVault' with your values.
# Use Get-AzLocation to get available locations and use the DisplayName.
# To use an existing resource group, comment out the line for New-AzResourceGroup

$Loc = 'MyLocation';
$KVRGname = 'MyKeyVaultResourceGroup';
$keyVaultName = 'MySecureVault';
New-AzResourceGroup -Name $KVRGname -Location $Loc;
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc;
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$keyVaultResourceId = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName
$KVRGname).ResourceId;
$diskEncryptionKeyVaultUrl = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName
$KVRGname).VaultUri;

# Step 2: Create the AD application and service principal.
# Fill in 'MyAADClientSecret', "<My Application Display Name>", "<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish.

$aadClientSecret = 'MyAADClientSecret';
$aadClientSecretSec = ConvertTo-SecureString -String $aadClientSecret -AsPlainText -Force;
$azureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "
<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -Password $aadClientSecretSec
$servicePrincipal = New-AzADServicePrincipal -ApplicationId $azureAdApplication.ApplicationId -Role
Contributor;
$aadClientID = $azureAdApplication.ApplicationId;

#Step 3: Enable the vault for disk encryption and set the access policy for the Azure AD application.

Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -
EnabledForDiskEncryption;
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -
PermissionsToKeys 'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname;

#Step 4: Create a new key in the key vault with the Add-AzKeyVaultKey cmdlet.
# Fill in 'MyKeyEncryptionKey' with your value.

$keyEncryptionKeyName = 'MyKeyEncryptionKey';
Add-AzKeyVaultKey -VaultName $keyVaultName -Name $keyEncryptionKeyName -Destination 'Software';
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name $keyEncryptionKeyName).Key.kid;

#Step 5: Encrypt the disks of an existing IaaS VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM';
$VMRGName = 'MyVirtualMachineResourceGroup';
Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID $aadClientID -
AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $keyVaultResourceId;
```

## Certificate-based authentication (optional)

If you would like to use certificate authentication, you can upload one to your key vault and deploy it to the client. Before using the PowerShell script, you should be familiar with the Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment.

```
# Fill in "MyKeyVaultResourceGroup", "MySecureVault", and 'MyLocation' ('My location' only if needed)
```

```

$KVRGname = 'MyKeyVaultResourceGroup'
$keyVaultName= 'MySecureVault'

# Create a key vault and set enabledForDiskEncryption property on it.
# Comment out the next three lines if you already have an existing key vault enabled for encryption. No
need to set 'My location' in this case.

$Loc = 'MyLocation'
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDiskEncryption

#Setting some variables with the key vault information
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname
$DiskEncryptionKeyVaultUrl = $keyVault.VaultUri
$keyVaultResourceId = $keyVault.ResourceId

# Create the Azure AD application and associate the certificate with it.
# Fill in "C:\certificates\mycert.pfx", "Password", "<My Application Display Name>", "
<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish

$CertPath = "C:\certificates\mycert.pfx"
$CertPassword = "Password"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath,
$CertPassword)
$CertValue = [System.Convert]::ToBase64String($cert.GetRawCertData())

$AzureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "
<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -CertValue $CertValue
$ServicePrincipal = New-AzADServicePrincipal -ApplicationId $AzureAdApplication.ApplicationId -Role
Contributor

$AADClientID = $AzureAdApplication.ApplicationId
$aadClientCertThumbprint= $cert.Thumbprint

Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -PermissionsToKeys
'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname

# Upload the pfx file to the key vault.
# Fill in "MyAADCert".

$keyVaultSecretName = "MyAADCert"
$fileContentBytes = get-content $CertPath -Encoding Byte
$fileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)
$jsonObject = @{
    "data" : "$filecontentencoded",
    "dataType" : "pfx",
    "password" : "$CertPassword"
}
"@

$jsonObjectBytes = [System.Text.Encoding]::UTF8.GetBytes($jsonObject)
$jsonEncoded = [System.Convert]::ToBase64String($jsonObjectBytes)

#Set the secret and set the key vault policy for -EnabledForDeployment

$secret = ConvertTo-SecureString -String $jsonEncoded -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName -SecretValue $secret
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDeployment

# Deploy the certificate to the VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM'
$VMRGName = 'MyVirtualMachineResourceGroup'
$CertUrl = (Get-AzKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName).Id
$SourceVaultId = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname).ResourceId
$VM = Get-AzVM -ResourceGroupName $VMRGName -Name $VMName

```

```
$VM = Add-AzVMSecret -VM $VM -SourceVaultId $SourceVaultId -CertificateStore "My" -CertificateUrl $CertUrl  
Update-AzVM -VM $VM -ResourceGroupName $VMRGName  
  
#Enable encryption on the VM using Azure AD client ID and the client certificate thumbprint  
  
Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -  
aadClientCertThumbprint $AADClientCertThumbprint -DiskEncryptionKeyVaultUrl $DiskEncryptionKeyVaultUrl -  
diskEncryptionKeyVaultId $KeyVaultResourceId
```

Certificate-based authentication and a KEK (optional)

If you would like to use certificate authentication and wrap the encryption key with a KEK, you can use the below script as an example. Before using the PowerShell script, you should be familiar with all of the previous Azure Disk Encryption prerequisites to understand the steps in the script. The sample script might need changes for your environment.

```

# Fill in 'MyKeyVaultResourceGroup', 'MySecureVault', and 'MyLocation' (if needed)

$KVRGname = 'MyKeyVaultResourceGroup'
$keyVaultName= 'MySecureVault'

# Create a key vault and set enabledForDiskEncryption property on it.
# Comment out the next three lines if you already have an existing key vault enabled for encryption.

$Loc = 'MyLocation'
New-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname -Location $Loc
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ResourceGroupName $KVRGname -EnabledForDiskEncryption

# Create the Azure AD application and associate the certificate with it.
# Fill in "C:\certificates\mycert.pfx", "Password", "<My Application Display Name>", "<https://MyApplicationHomePage>", and "<https://MyApplicationUri>" with your values.
# MyApplicationHomePage and the MyApplicationUri can be any values you wish

$CertPath = "C:\certificates\mycert.pfx"
$CertPassword = "Password"
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath,
$CertPassword)
$CertValue = [System.Convert]::ToBase64String($cert.GetRawCertData())

$AzureAdApplication = New-AzADApplication -DisplayName "<My Application Display Name>" -HomePage "<https://MyApplicationHomePage>" -IdentifierUris "<https://MyApplicationUri>" -CertValue $CertValue
$ServicePrincipal = New-AzADServicePrincipal -ApplicationId $AzureAdApplication.ApplicationId -Role
Contributor

$AADClientID = $AzureAdApplication.ApplicationId
$aadClientCertThumbprint= $cert.Thumbprint

## Give access for setting secrets and wrapping keys
Set-AzKeyVaultAccessPolicy -VaultName $keyVaultName -ServicePrincipalName $aadClientID -PermissionsToKeys
'WrapKey' -PermissionsToSecrets 'Set' -ResourceGroupName $KVRGname

# Upload the pfx file to the key vault.
# Fill in "MyAADCert".

$keyVaultSecretName = "MyAADCert"
$FileContentBytes = get-content $CertPath -Encoding Byte
$FileContentEncoded = [System.Convert]::ToBase64String($fileContentBytes)
    $JSONObject = @"
{
    "data" : "$filecontentencoded",
    "dataType" : "pfx",
    "password" : "$CertPassword"
}
"@
```

```

$JSONObjectBytes = [System.Text.Encoding]::UTF8.GetBytes($jsonObject)
$JSONEncoded = [System.Convert]::ToBase64String($jsonObjectBytes)

#Set the secret and set the key vault policy for deployment

$Secret = ConvertTo-SecureString -String $JSONEncoded -AsPlainText -Force
Set-AzKeyVaultSecret -VaultName $KeyVaultName -Name $KeyVaultSecretName -SecretValue $Secret
Set-AzKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $KVRGname -EnabledForDeployment

#Setting some variables with the key vault information and generating a KEK
# Fill in 'KEKName'

$KEKName ='KEKName'
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname
$DiskEncryptionKeyVaultUrl = $keyVault.VaultUri
$keyVaultResourceId = $keyVault.ResourceId
$KEK = Add-AzKeyVaultKey -VaultName $keyVaultName -Name $KEKName -Destination "Software"
$keyEncryptionKeyUrl = $KEK.Key.kid


# Deploy the certificate to the VM
# Fill in 'MySecureVM' and 'MyVirtualMachineResourceGroup' with your values.

$VMName = 'MySecureVM';
$VMRGName = 'MyVirtualMachineResourceGroup';
$CertUrl = (Get-AzKeyVaultSecret -VaultName $keyVaultName -Name $keyVaultSecretName).Id
$SourceVaultId = (Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGName).ResourceId
$VM = Get-AzVM -ResourceGroupName $VMRGName -Name $VMName
$VM = Add-AzVMSecret -VM $VM -SourceVaultId $SourceVaultId -CertificateStore "My" -CertificateUrl $CertUrl
Update-AzVM -VM $VM -ResourceGroupName $VMRGName

#Enable encryption on the VM using Azure AD client ID and the client certificate thumbprint

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -
AadClientCertThumbprint $AADClientCertThumbprint -DiskEncryptionKeyVaultUrl $DiskEncryptionKeyVaultUrl -
DiskEncryptionKeyId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyId $keyVaultResourceId

```

## Next steps

[Enable Azure Disk Encryption with Azure AD on Windows VMs \(previous release\)](#)

# Azure Disk Encryption with Azure AD for Windows VMs (previous release)

9/21/2022 • 13 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The new release of Azure Disk Encryption eliminates the requirement for providing an Azure AD application parameter to enable VM disk encryption. With the new release, you are no longer required to provide Azure AD credentials during the enable encryption step. All new VMs must be encrypted without the Azure AD application parameters using the new release. To view instructions to enable VM disk encryption using the new release, see [Azure Disk Encryption for Windows VMs](#). VMs that were already encrypted with Azure AD application parameters are still supported and should continue to be maintained with the AAD syntax.

You can enable many disk-encryption scenarios, and the steps may vary according to the scenario. The following sections cover the scenarios in greater detail for Windows IaaS VMs. Before you can use disk encryption, the [Azure Disk Encryption prerequisites](#) need to be completed.

## IMPORTANT

- You should [take a snapshot](#) and/or create a backup before disks are encrypted. Backups ensure that a recovery option is possible if an unexpected failure occurs during encryption. VMs with managed disks require a backup before encryption occurs. Once a backup is made, you can use the [Set-AzVMDiskEncryptionExtension cmdlet](#) to encrypt managed disks by specifying the `-skipVmBackup` parameter. For more information about how to back up and restore encrypted VMs, see [Back up and restore encrypted Azure VM](#).
- Encrypting or disabling encryption may cause a VM to reboot.

## Enable encryption on new IaaS VMs created from the Marketplace

You can enable disk encryption on new IaaS Windows VM from the Marketplace in Azure using a Resource Manager template. The template creates a new encrypted Windows VM using the Windows Server 2012 gallery image.

1. On the [Resource Manager template](#), click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, parameters, legal terms, and agreement. Click **Purchase** to deploy a new IaaS VM where encryption is enabled.
3. After you deploy the template, verify the VM encryption status using your preferred method:
  - Verify with the Azure CLI by using the [az vm encryption show](#) command.

```
az vm encryption show --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup"
```

- Verify with Azure PowerShell by using the [Get-AzVmDiskEncryptionStatus cmdlet](#).

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

- Select the VM, then click on **Disks** under the **Settings** heading to verify encryption status in the portal. In the chart under **Encryption**, you'll see if it's enabled.

NAME	STATUS	LOCATION	SIZE	DISK ENCRYPTION
ADEDemoCAT	Running	Australia East	Standard_D1	Enabled
ADEPreDemoCAT	Running	Australia East	Standard_D1	Enabled
at-east	Running	East US	Standard_A1	Enabled
at-prevm10	Running	Australia East	Standard_D2	Enabled

The following table lists the Resource Manager template parameters for new VMs from the Marketplace scenario using Azure AD client ID:

PARAMETER	DESCRIPTION
adminUserName	Admin user name for the virtual machine.
adminPassword	Admin user password for the virtual machine.
newStorageAccountName	Name of the storage account to store OS and data VHDs.
vmSize	Size of the VM. Currently, only Standard A, D, and G series are supported.
virtualNetworkName	Name of the VNet that the VM NIC should belong to.
subnetName	Name of the subnet in the VNet that the VM NIC should belong to.
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to your key vault.
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to your key vault.
keyVaultURL	<p>URL of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet</p> <pre>(Get-AzKeyVault -VaultName "MyKeyVault" -ResourceGroupName "MyKeyVaultResourceGroupName").VaultURI</pre> <p>or the Azure CLI</p> <pre>az keyvault show --name "MySecureVault" --query properties.vaultUri</pre>
keyEncryptionKeyURL	<p>URL of the key encryption key that's used to encrypt the generated BitLocker key (optional).</p> <p>KeyEncryptionKeyURL is an optional parameter. You can bring your own KEK to further safeguard the data encryption key (Passphrase secret) in your key vault.</p>
keyVaultResourceGroup	Resource group of the key vault.

PARAMETER	DESCRIPTION
vmName	Name of the VM that the encryption operation is to be performed on.

## Enable encryption on existing or running IaaS Windows VMs

In this scenario, you can enable encryption by using a template, PowerShell cmdlets, or CLI commands. The following sections explain in greater detail how to enable Azure Disk Encryption.

### Enable encryption on existing or running VMs with Azure PowerShell

Use the [Set-AzVMDiskEncryptionExtension](#) cmdlet to enable encryption on a running IaaS virtual machine in Azure. For information about enabling encryption with Azure Disk Encryption by using PowerShell cmdlets, see the blog posts [Explore Azure Disk Encryption with Azure PowerShell - Part 1](#) and [Explore Azure Disk Encryption with Azure PowerShell - Part 2](#).

- **Encrypt a running VM using a client secret:** The script below initializes your variables and runs the Set-AzVMDiskEncryptionExtension cmdlet. The resource group, VM, key vault, AAD app, and client secret should have already been created as prerequisites. Replace MyKeyVaultResourceGroup, MyVirtualMachineResourceGroup, MySecureVM, MySecureVault, My-AAD-client-ID, and My-AAD-client-secret with your values.

```
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId;
```

- **Encrypt a running VM using KEK to wrap the client secret:** Azure Disk Encryption lets you specify an existing key in your key vault to wrap disk encryption secrets that were generated while enabling encryption. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault.

```

$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $keyVaultResourceId;

```

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

```
/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-
name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]
```

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-
name].vault.azure.net/keys/[kekname]/[kek-unique-id]

- **Verify the disks are encrypted:** To check on the encryption status of an IaaS VM, use the [Get-AzVmDiskEncryptionStatus](#) cmdlet.

```
Get-AzVmDiskEncryptionStatus -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

- **Disable disk encryption:** To disable the encryption, use the [Disable-AzureRmVMDiskEncryption](#) cmdlet.

```
Disable-AzVMDiskEncryption -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

### Enable encryption on existing or running VMs with Azure CLI

Use the [az vm encryption enable](#) command to enable encryption on a running IaaS virtual machine in Azure.

- **Encrypt a running VM using a client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-
client-id "<my spn created with CLI/my Azure AD ClientID>" --aad-client-secret "My-AAD-client-
secret" --disk-encryption-keyvault "MySecureVault" --volume-type [All|OS|Data]
```

- **Encrypt a running VM using KEK to wrap the client secret:**

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-
client-id "<my spn created with CLI which is the Azure AD ClientID>" --aad-client-secret "My-AAD-
client-secret" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-
encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type [All|OS|Data]
```

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:

```
/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]
```

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]

- **Verify the disks are encrypted:** To check on the encryption status of an IaaS VM, use the [az vm encryption show](#) command.

```
az vm encryption show --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup"
```

- **Disable encryption:** To disable encryption, use the [az vm encryption disable](#) command.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --volume-type [ALL, DATA, OS]
```

## Using the Resource Manager template

You can enable disk encryption on existing or running IaaS Windows VMs in Azure by using the [Resource Manager template to encrypt a running Windows VM](#).

1. On the Azure quickstart template, click **Deploy to Azure**.
2. Select the subscription, resource group, resource group location, parameters, legal terms, and agreement. Click **Purchase** to enable encryption on the existing or running IaaS VM.

The following table lists the Resource Manager template parameters for existing or running VMs that use an Azure AD client ID:

PARAMETER	DESCRIPTION
AADClientID	Client ID of the Azure AD application that has permissions to write secrets to the key vault.
AADClientSecret	Client secret of the Azure AD application that has permissions to write secrets to the key vault.
keyVaultName	Name of the key vault that the BitLocker key should be uploaded to. You can get it by using the cmdlet <pre>(Get-AzKeyVault -ResourceGroupName &lt;MyKeyVaultResourceGroupName&gt;).Vaultname</pre> or the Azure CLI command <pre>az keyvault list --resource-group "MySecureGroup"</pre>
keyEncryptionKeyURL	URL of the key encryption key that's used to encrypt the generated BitLocker key. This parameter is optional if you select <b>nokek</b> in the UseExistingKek drop-down list. If you select <b>kek</b> in the UseExistingKek drop-down list, you must enter the <i>keyEncryptionKeyURL</i> value.
volumeType	Type of volume that the encryption operation is performed on. Valid values are <i>OS</i> , <i>Data</i> , and <i>All</i> .

PARAMETER	DESCRIPTION
sequenceVersion	Sequence version of the BitLocker operation. Increment this version number every time a disk-encryption operation is performed on the same VM.
vmName	Name of the VM that the encryption operation is to be performed on.

## New IaaS VMs created from customer-encrypted VHD and encryption keys

In this scenario, you can enable encrypting by using the Resource Manager template, PowerShell cmdlets, or CLI commands. The following sections explain in greater detail the Resource Manager template and CLI commands.

Use the instructions in the appendix for preparing pre-encrypted images that can be used in Azure. After the image is created, you can use the steps in the next section to create an encrypted Azure VM.

- [Prepare a pre-encrypted Windows VHD](#)

### Encrypt VMs with pre-encrypted VHDS with Azure PowerShell

You can enable disk encryption on your encrypted VHD by using the PowerShell cmdlet [Set-AzVMOSDisk](#). The example below gives you some common parameters.

```
$VirtualMachine = New-AzVMConfig -VMName "MySecureVM" -VMSize "Standard_A1"
$VirtualMachine = Set-AzVMOSDisk -VM $VirtualMachine -Name "SecureOSDisk" -VhdUri "os.vhd" Caching ReadWrite
-Windows -CreateOption "Attach" -DiskEncryptionKeyUrl
"https://mytestvault.azure.net/secrets/Test1/514ceb769c984379a7e0230bddaaaaaa" -
DiskEncryptionKeyVaultId "/subscriptions/00000000-0000-0000-0000-
0000000000/resourceGroups/myKvresourcegroup/providers/Microsoft.KeyVault/vaults/mytestvault"
New-AzVM -VM $VirtualMachine -ResourceGroupName "MyVirtualMachineResourceGroup"
```

## Enable encryption on a newly added data disk

You can [add a new disk to a Windows VM using PowerShell](#), or [through the Azure portal](#).

### Enable encryption on a newly added disk with Azure PowerShell

When using PowerShell to encrypt a new disk for Windows VMs, a new sequence version should be specified. The sequence version has to be unique. The script below generates a GUID for the sequence version. In some cases, a newly added data disk might be encrypted automatically by the Azure Disk Encryption extension. Auto encryption usually occurs when the VM reboots after the new disk comes online. This is typically caused because "All" was specified for the volume type when disk encryption previously ran on the VM. If auto encryption occurs on a newly added data disk, we recommend running the [Set-AzVmDiskEncryptionExtension](#) cmdlet again with new sequence version. If your new data disk is auto encrypted and you do not wish to be encrypted, decrypt all drives first then re-encrypt with a new sequence version specifying OS for the volume type.

- **Encrypt a running VM using a client secret:** The script below initializes your variables and runs the [Set-AzVMDiskEncryptionExtension](#) cmdlet. The resource group, VM, key vault, AAD app, and client secret should have already been created as prerequisites. Replace MyKeyVaultResourceGroup, MyVirtualMachineResourceGroup, MySecureVM, MySecureVault, My-AAD-client-ID, and My-AAD-client-secret with your values. This example uses "All" for the -VolumeType parameter, which includes both OS and Data volumes. If you only want to encrypt the OS volume, use "OS" for the -VolumeType parameter.

```

$sequenceVersion = [Guid]::.NewGuid();
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MySecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId -VolumeType 'all' -SequenceVersion $sequenceVersion;

```

- Encrypt a running VM using KEK to wrap the client secret: Azure Disk Encryption lets you specify an existing key in your key vault to wrap disk encryption secrets that were generated while enabling encryption. When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault. This example uses "All" for the -VolumeType parameter, which includes both OS and Data volumes. If you only want to encrypt the OS volume, use "OS" for the -VolumeType parameter.

```

$sequenceVersion = [Guid]::.NewGuid();
$KVRGname = 'MyKeyVaultResourceGroup';
$VMRGName = 'MyVirtualMachineResourceGroup';
$vmName = 'MyExtraSecureVM';
$aadClientID = 'My-AAD-client-ID';
$aadClientSecret = 'My-AAD-client-secret';
$keyVaultName = 'MySecureVault';
$keyEncryptionKeyName = 'MyKeyEncryptionKey';
$keyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $keyVault.VaultUri;
$keyVaultResourceId = $keyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name
$keyEncryptionKeyName).Key.kid;

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $vmName -AadClientID
$aadClientID -AadClientSecret $aadClientSecret -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl
-DiskEncryptionKeyVaultId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $keyVaultResourceId -VolumeType 'all' -SequenceVersion $sequenceVersion;

```

#### NOTE

The syntax for the value of disk-encryption-keyvault parameter is the full identifier string:  
`/subscriptions/[subscription-id-guid]/resourceGroups/[resource-group-name]/providers/Microsoft.KeyVault/vaults/[keyvault-name]`

The syntax for the value of the key-encryption-key parameter is the full URI to the KEK as in: `https://[keyvault-name].vault.azure.net/keys/[kekname]/[kek-unique-id]`

## Enable encryption on a newly added disk with Azure CLI

The Azure CLI command will automatically provide a new sequence version for you when you run the command to enable encryption. Acceptable values for the volume-type parameter are All, OS, and Data. You may need to change the volume-type parameter to OS or Data if you're only encrypting one type of disk for the VM. The examples use "All" for the volume-type parameter.

- Encrypt a running VM using a client secret:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI/my Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --volume-type "All"
```

- Encrypt a running VM using KEK to wrap the client secret:

```
az vm encryption enable --resource-group "MyVirtualMachineResourceGroup" --name "MySecureVM" --aad-client-id "<my spn created with CLI which is the Azure AD ClientID>" --aad-client-secret "My-AAD-client-secret" --disk-encryption-keyvault "MySecureVault" --key-encryption-key "MyKEK_URI" --key-encryption-keyvault "MySecureVaultContainingTheKEK" --volume-type "all"
```

## Enable encryption using Azure AD client certificate-based authentication.

You can use client certificate authentication with or without KEK. Before using the PowerShell scripts, you should already have the certificate uploaded to the key vault and deployed to the VM. If you're using KEK too, the KEK should already exist. For more information, see the [Certificate-based authentication for Azure AD](#) section of the prerequisites article.

### Enable encryption using certificate-based authentication with Azure PowerShell

```
## Fill in 'MyVirtualMachineResourceGroup', 'MyKeyVaultResourceGroup', 'My-AAD-client-ID', 'MySecureVault', and 'MySecureVM'.
```

```
$VMRGName = 'MyVirtualMachineResourceGroup'
$KVRGname = 'MyKeyVaultResourceGroup';
$AADClientID ='My-AAD-client-ID';
$keyVaultName = 'MySecureVault';
$VMName = 'MySecureVM';
$KeyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $KeyVault.VaultUri;
$keyVaultResourceId = $KeyVault.ResourceId;

# Fill in the certificate path and the password so the thumbprint can be set as a variable.

$certPath = '$CertPath = "C:\certificates\mycert.pfx";
$CertPassword ='Password'
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath, $CertPassword)
$aadClientCertThumbprint = $cert.Thumbprint;

# Enable disk encryption using the client certificate thumbprint

Set-AzMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -
AadClientCertThumbprint $aadClientCertThumbprint -DiskEncryptionKeyVaultUrl $DiskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $keyVaultResourceId
```

### Enable encryption using certificate-based authentication and a KEK with Azure PowerShell

```

# Fill in 'MyVirtualMachineResourceGroup', 'MyKeyVaultResourceGroup', 'My-AAD-client-ID', 'MySecureVault,,,'
'MySecureVM', and "KEKName.

$VMRGName = 'MyVirtualMachineResourceGroup';
$KVRGname = 'MyKeyVaultResourceGroup';
$AADClientID ='My-AAD-client-ID';
$keyVaultName = 'MySecureVault';
$VMName = 'MySecureVM';
$keyEncryptionKeyName = 'KEKName';
$KeyVault = Get-AzKeyVault -VaultName $keyVaultName -ResourceGroupName $KVRGname;
$diskEncryptionKeyVaultUrl = $KeyVault.VaultUri;
$keyVaultResourceId = $KeyVault.ResourceId;
$keyEncryptionKeyUrl = (Get-AzKeyVaultKey -VaultName $keyVaultName -Name $keyEncryptionKeyName).Key.kid;

## Fill in the certificate path and the password so the thumbprint can be read and set as a variable.

$certPath = '$CertPath = "C:\certificates\mycert.pfx";
$CertPassword ='Password'
$Cert = New-Object System.Security.Cryptography.X509Certificates.X509Certificate2($CertPath, $CertPassword)
$aadClientCertThumbprint = $cert.Thumbprint;

# Enable disk encryption using the client certificate thumbprint and a KEK

Set-AzVMDiskEncryptionExtension -ResourceGroupName $VMRGName -VMName $VMName -AadClientID $AADClientID -
AadClientCertThumbprint $aadClientCertThumbprint -DiskEncryptionKeyVaultUrl $diskEncryptionKeyVaultUrl -
DiskEncryptionKeyVaultId $keyVaultResourceId -KeyEncryptionKeyUrl $keyEncryptionKeyUrl -
KeyEncryptionKeyVaultId $keyVaultResourceId

```

## Disable encryption

You can disable encryption using Azure PowerShell, the Azure CLI, or with a Resource Manager template.

- **Disable disk encryption with Azure PowerShell:** To disable the encryption, use the [Disable-AzureRmVMDiskEncryption](#) cmdlet.

```
Disable-AzVMDiskEncryption -ResourceGroupName 'MyVirtualMachineResourceGroup' -VMName 'MySecureVM'
```

- **Disable encryption with the Azure CLI:** To disable encryption, use the [az vm encryption disable](#) command.

```
az vm encryption disable --name "MySecureVM" --resource-group "MyVirtualMachineResourceGroup" --
volume-type [ALL, DATA, OS]
```

- **Disable encryption with a Resource Manager Template:**

1. Click **Deploy to Azure** from the [Disable disk encryption on running Windows VM](#) template.
2. Select the subscription, resource group, location, VM, legal terms, and agreement.
3. Click **Purchase** to disable disk encryption on a running Windows VM.

## Next steps

[Azure Disk Encryption overview](#)

# Using Azure ultra disks

9/21/2022 • 14 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article explains how to deploy and use an ultra disk, for conceptual information about ultra disks, refer to [What disk types are available in Azure?](#).

Azure ultra disks offer high throughput, high IOPS, and consistent low latency disk storage for Azure IaaS virtual machines (VMs). This new offering provides top of the line performance at the same availability levels as our existing disks offerings. One major benefit of ultra disks is the ability to dynamically change the performance of the SSD along with your workloads without the need to restart your VMs. Ultra disks are suited for data-intensive workloads such as SAP HANA, top tier databases, and transaction-heavy workloads.

## GA scope and limitations

Ultra disks can't be used as OS disks, they can only be created as empty data disks. Ultra disks also can't be used with some features and functionality, including disk snapshots, disk export, changing disk type, VM images, availability sets, Azure Dedicated Hosts, or Azure disk encryption. Azure Backup and Azure Site Recovery do not support ultra disks. In addition, only un-cached reads and un-cached writes are supported.

Ultra disks support a 4k physical sector size by default. A 512E sector size is available as a generally available offering with no sign-up required. While most applications are compatible with 4k sector sizes, some require 512 byte sector sizes. Oracle Database, for example, requires release 12.2 or later in order to support 4k native disks. For older versions of Oracle DB, 512 byte sector size is required.

The only infrastructure redundancy options currently available to ultra disks are availability zones. VMs using any other redundancy options cannot attach an ultra disk.

The following table outlines the regions ultra disks are available in, as well as their corresponding availability options.

### NOTE

If a region in the following list lacks availability zones that support ultra disks, then a VM in that region must be deployed without infrastructure redundancy in order to attach an ultra disk.

REDUNDANCY OPTIONS	REGIONS
Single VMs	Australia Central Brazil South Central India East Asia Germany West Central Korea Central North Central US, South Central US, West US US Gov Arizona, US Gov Texas, US Gov Virginia
One availability zone	China North 3 Qatar Central
Two availability zones	France Central

REDUNDANCY OPTIONS	REGIONS
Three availability zones	Australia East Canada Central North Europe, West Europe Japan East Southeast Asia Sweden Central UK South Central US, East US, East US 2, West US 2, West US 3

Not every VM size is available in every supported region with ultra disks. The following table lists VM series which are compatible with ultra disks.

VM TYPE	SIZES	DESCRIPTION
General purpose	<a href="#">DSv3-series</a> , <a href="#">Ddsv4-series</a> , <a href="#">Dsv4-series</a> , <a href="#">Dasv4-series</a> , <a href="#">Dsv5-series</a> , <a href="#">Ddsv5-series</a> , <a href="#">Dasv5-series</a>	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	<a href="#">FSv2-series</a>	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	<a href="#">ESv3-series</a> , <a href="#">Easv4-series</a> , <a href="#">Edsv4-series</a> , <a href="#">Esv4-series</a> , <a href="#">Esv5-series</a> , <a href="#">Edsv5-series</a> , <a href="#">Easv5-series</a> , <a href="#">Ebsv5 series</a> , <a href="#">Ebdsv5 series</a> , <a href="#">M-series</a> , <a href="#">Mv2-series</a> , <a href="#">Msv2/Mdsv2-series</a>	High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	<a href="#">LSv2-series</a>	High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases.
GPU optimized	<a href="#">NCv2-series</a> , <a href="#">NCv3-series</a> , <a href="#">NCasT4_v3-series</a> , <a href="#">ND-series</a> , <a href="#">NDv2-series</a> , <a href="#">NVv3-series</a> , <a href="#">NVv4-series</a> , <a href="#">NVadsA10 v5-series</a>	Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs.
Performance optimized	<a href="#">HB-series</a> , <a href="#">HC-series</a> , <a href="#">HBv2-series</a>	The fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

## Determine VM size and region availability

### VMs using availability zones

To leverage ultra disks, you need to determine which availability zone you are in. Not every region supports every VM size with ultra disks. To determine if your region, zone, and VM size support ultra disks, run either of the following commands, make sure to replace the `region`, `vmSize`, and `subscription` values first:

**CLI**

```

subscription=<yourSubID>
# example value is southeastasia
region=<yourLocation>
# example value is Standard_E64s_v3
vmSize=<yourVMSize>

az vm list-skus --resource-type virtualMachines --location $region --query "[?name=='$vmSize'].locationInfo[0].zoneDetails[0].Name" --subscription $subscription

```

## PowerShell

```

$region = "southeastasia"
$vmSize = "Standard_E64s_v3"
$sku = (Get-AzComputeResourceSku | where {$_.Locations.Contains($region) -and ($_.Name -eq $vmSize) -and $_.LocationInfo[0].ZoneDetails.Count -gt 0})
if($sku){$sku[0].LocationInfo[0].ZoneDetails} Else {Write-host "$vmSize is not supported with Ultra Disk in $region region"}

```

The response will be similar to the form below, where X is the zone to use for deploying in your chosen region. X could be either 1, 2, or 3.

Preserve the **Zones** value, it represents your availability zone and you will need it in order to deploy an Ultra disk.

RESOURCE TYPE	NAME	LOCATION	ZONES	RESTRICTION	CAPABILITY	VALUE
disks	UltraSSD_LRS	eastus2	X			

### NOTE

If there was no response from the command, then the selected VM size is not supported with ultra disks in the selected region.

Now that you know which zone to deploy to, follow the deployment steps in this article to either deploy a VM with an ultra disk attached or attach an ultra disk to an existing VM.

### VMs with no redundancy options

Ultra disks deployed in select regions must be deployed without any redundancy options, for now. However, not every disk size that supports ultra disks may be in these region. To determine which disk sizes support ultra disks, you can use either of the following code snippets. Make sure to replace the `vmSize` and `subscription` values first:

```

subscription=<yourSubID>
region="westus"
# example value is Standard_E64s_v3
vmSize=<yourVMSize>

az vm list-skus --resource-type virtualMachines --location $region --query "[?name=='$vmSize'].capabilities" --subscription $subscription

```

```

$region = "westus"
$vmSize = "Standard_E64s_v3"
(Get-AzComputeResourceSku | where {$_.Locations.Contains($region) -and ($_.Name -eq $vmSize) })[0].Capabilities

```

The response will be similar to the following form, `UltraSSDAvailable True` indicates whether the VM size supports ultra disks in this region.

Name	Value
---	----
MaxResourceVolumeMB	884736
OSVhdSizeMB	1047552
vCPUs	64
HyperVGenerations	V1, V2
MemoryGB	432
MaxDataDiskCount	32
LowPriorityCapable	True
PremiumIO	True
VMDeploymentTypes	IaaS
vCPUsAvailable	64
ACUs	160
vCPUsPerCore	2
CombinedTempDiskAndCachedIOPS	128000
CombinedTempDiskAndCachedReadBytesPerSecond	1073741824
CombinedTempDiskAndCachedWriteBytesPerSecond	1073741824
CachedDiskBytes	1717986918400
UncachedDiskIOPS	80000
UncachedDiskBytesPerSecond	1258291200
EphemeralOSDiskSupported	True
AcceleratedNetworkingEnabled	True
RdmaEnabled	False
MaxNetworkInterfaces	8
UltraSSDAvailable	True

## Deploy an ultra disk using Azure Resource Manager

First, determine the VM size to deploy. For a list of supported VM sizes, see the [GA scope and limitations](#) section.

If you would like to create a VM with multiple ultra disks, refer to the sample [Create a VM with multiple ultra disks](#).

If you intend to use your own template, make sure that `apiVersion` for `Microsoft.Compute/virtualMachines` and `Microsoft.Compute/Disks` is set as `2018-06-01` (or later).

Set the disk sku to `UltraSSD_LRS`, then set the disk capacity, IOPS, availability zone, and throughput in MBps to create an ultra disk.

Once the VM is provisioned, you can partition and format the data disks and configure them for your workloads.

## Deploy an ultra disk

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)

This section covers deploying a virtual machine equipped with an ultra disk as a data disk. It assumes you have familiarity with deploying a virtual machine, if you do not, see our [Quickstart: Create a Windows virtual machine in the Azure portal](#).

1. Sign in to the [Azure portal](#) and navigate to deploy a virtual machine (VM).
2. Make sure to choose a [supported VM size and region](#).
3. Select **Availability zone** in **Availability options**.

4. Fill in the remaining entries with selections of your choice.

5. Select Disks.

The screenshot shows the 'Create a virtual machine' wizard in the Microsoft Azure portal. The 'Basics' tab is selected. The 'Instance details' section is filled out with the following values:

- Virtual machine name: myVMName
- Region: (US) West US 2
- Availability options: Availability zone
- Availability zone: 1
- Image: Ubuntu Server 18.04 LTS - Gen1
- Size: Standard\_D2s\_v3 - 2 vcpus, 8 GiB memory (\$24.50/month)

The 'Disks' tab is highlighted with a red border. Below it, the 'Review + create' button is visible.

6. On the Disks blade, select Yes for Enable Ultra Disk compatibility.

7. Select Create and attach a new disk to attach an ultra disk now.

The screenshot shows the 'Disks' blade in the Microsoft Azure portal. The 'Disk options' section includes:

- OS disk type: Premium SSD
- Enable Ultra Disk compatibility:

The 'Data disks' section displays a table with the following columns: LUN, Name, Size (GiB), Disk type, and Host caching. At the bottom of the blade, there are two buttons: 'Create and attach a new disk' and 'Attach an existing disk'.

8. On the Create a new disk blade, enter a name, then select Change size.

## Create a new disk ...

Create a new disk to store applications and data on your VM. Disk pricing varies based on factors including disk size, storage type, and number of transactions. [Learn more ↗](#)

Name *	myVMName_DataDisk_0
Source type *	None (empty disk)
Size *	<b>1024 GiB</b> Premium SSD LRS <a href="#">Change size</a>
Encryption type *	(Default) Encryption at-rest with a platform-managed key
Enable shared disk	<input type="radio"/> Yes <input checked="" type="radio"/> No

9. Change the Disk SKU to Ultra Disk.

10. Change the values of Custom disk size (GiB), Disk IOPS, and Disk throughput to ones of your choice.

11. Select OK in both blades.

### Select a disk size ...

Browse available disk sizes and their features.

Disk SKU	Max size	Performance tier	Provisioned IOPS	Provisioned through...	Max Shares	Max burst IOPS	Max burst throughput
Ultra Disk (locally-redundant storage)	65536 GiB	U	160000	2000	5	-	-

Custom disk size (GiB) \* 1024

Disk IOPS \* 2048

Disk throughput (MB/s) \* 8

12. Continue with the VM deployment, it will be the same as you would deploy any other VM.

## Deploy an ultra disk - 512 byte sector size

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)

1. Sign in to the [Azure portal](#), then search for and select Disks.

2. Select + New to create a new disk.

3. Select a region that supports ultra disks and select an availability zone, fill in the rest of the values as you desire.

4. Select **Change size**.

## Create a managed disk

Basics    Encryption    Networking    Advanced    Tags    Review + create

Select the disk type and size needed for your workload. Azure disks are designed for 99.999% availability. Azure managed disks encrypt your data at rest, by default, using Storage Service Encryption. [Learn more about disks.](#)

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	<input type="text" value="My Example Subscription"/>
Resource group *	<input type="text"/> <a href="#">Create new</a>

### Disk details

Disk name *	<input type="text"/>
Region *	<input type="text" value="(US) West US 2"/>
Availability zone	<input type="text" value="1"/>
Source type	<input type="text" value="None"/>
Size *	<b>1024 GiB</b> Premium SSD LRS <a href="#">Change size</a>

5. For Disk SKU select Ultra disk, then fill in the values for the desired performance and select OK.

## Select a disk size

Browse available disk sizes and their features.

Disk SKU	<input type="text" value="Ultra Disk (locally-redundant storage)"/>
Max size	Performance tier
65536 GiB	U
Custom disk size (GiB) *	<input type="text" value="1024"/>
Disk IOPS *	<input type="text" value="2048"/>
Disk throughput (MB/s) *	<input type="text" value="8"/>

6. On the Basics blade, select the Advanced tab.

7. Select 512 for Logical sector size, then select Review + Create.

## Create a managed disk

Basics   Encryption   Networking   **Advanced**   Tags   Review + create

Add additional configurations for your managed disk

### Shared disk

Allow this disk to be attached to two or more virtual machines, depending on storage type and disk size. When shared disk is enabled host caching is unavailable. [Learn more about shared disks](#)

Enable shared disk    Yes    No

### Ultra disk

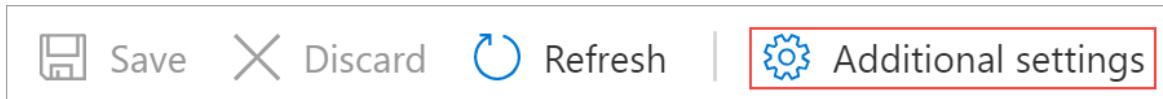
Logical sector size (bytes)    4096    512

## Attach an ultra disk

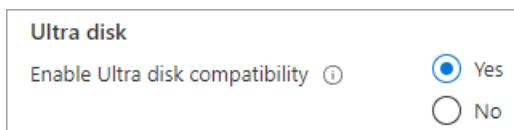
- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)

Alternatively, if your existing VM is in a region/availability zone that is capable of using ultra disks, you can make use of ultra disks without having to create a new VM. By enabling ultra disks on your existing VM, then attaching them as data disks. To enable ultra disk compatibility, you must stop the VM. After you stop the VM, you may enable compatibility, then restart the VM. Once compatibility is enabled you can attach an ultra disk:

1. Navigate to your VM and stop it, wait for it to deallocate.
2. Once your VM has been deallocated, select **Disks**.
3. Select **Additional settings**.



4. Select **Yes** for **Enable Ultra Disk compatibility**.



5. Select **Save**.
6. Select **Create and attach a new disk** and fill in a name for your new disk.
7. For **Storage type** select **Ultra Disk**.
8. Change the values of **Size (GiB)**, **Max IOPS**, and **Max throughput** to ones of your choice.
9. After you are returned to your disk's blade, select **Save**.

LUN	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (MBps)	Encryption	Host caching
0	my-ultra-disk	Ultra Disk	4	120	25	Platform-managed key	None

10. Start your VM again.

## Adjust the performance of an ultra disk

- [Portal](#)
- [Azure CLI](#)
- [PowerShell](#)

Ultra disks offer a unique capability that allows you to adjust their performance. You can make these adjustments from the Azure portal, on the disks themselves.

1. Navigate to your VM and select Disks.
2. Select the ultra disk you'd like to modify the performance of.

LUN	Disk name	Storage type
0	ultra-disk-name	Ultra disk LRS

**Create and attach a new disk**   **Attach existing disks**

3. Select **Size + performance** and then make your modifications.

4. Select **Save**.

ultra-disk-name | Size + performance

Disk

Search (Ctrl+ /)

Disk SKU ⓘ

Ultra Disk (locally-redundant storage)

Max size

65536 GiB

Performance tier

U

Provisioned IOPS

160000

Custom disk size (GiB) \*

4

Disk IOPS \*

120

Disk throughput (MB/s) \*

25

## Next steps

- [Use Azure ultra disks on Azure Kubernetes Service \(preview\).](#)
- [Migrate log disk to an ultra disk.](#)

# Deploy a Premium SSD v2 (preview)

9/21/2022 • 4 minutes to read • [Edit Online](#)

Azure Premium SSD v2 (preview) is designed for IO-intense enterprise workloads that require sub-millisecond disk latencies and high IOPS and throughput at a low cost. Premium SSD v2 is suited for a broad range of workloads such as SQL server, Oracle, MariaDB, SAP, Cassandra, Mongo DB, big data/analytics, gaming, on virtual machines or stateful containers. For conceptual information on Premium SSD v2, see [Premium SSD v2 \(preview\)](#).

## Limitations

- Premium SSD v2 disks can't be used as an OS disk.
- Currently, Premium SSD v2 disks can only be attached to zonal VMs.
- Currently, Premium SSD v2 disks can't be attached to VMs in virtual machine scale sets.
- Currently, taking snapshots aren't supported, and you can't create a Premium SSD v2 from the snapshot of another disk type.
- Currently, Premium SSD v2 disks can't be attached to VMs with encryption at host enabled.
- Currently, Premium SSD v2 disks can't be attached to VMs in Availability Sets.
- Azure Disk Encryption isn't supported for VMs with Premium SSD v2 disks.
- Azure Backup and Azure Site Recovery aren't supported for VMs with Premium SSD v2 disks.

## Regional availability

Currently only available in the following regions:

- US East
- West Europe

## Prerequisites

- [Sign-up](#) for the public preview.
- Install either the latest [Azure CLI](#) or the latest [Azure PowerShell module](#).

## Determine region availability programmatically

To use a Premium SSD v2, you need to determine the regions and zones where it's supported. Not every region and zones support Premium SSD v2. To determine regions, and zones support premium SSD v2, replace

`yourSubscriptionId` then run the following command:

- [Azure CLI](#)
- [PowerShell](#)
- [Azure portal](#)

```

az login

subscriptionId=<yourSubscriptionId>

az account set --subscription $subscriptionId

az vm list-skus --resource-type disks --query "[?name=='PremiumV2_LRS'].{Region:locationInfo[0].location,
Zones:locationInfo[0].zones}"

```

Now that you know the region and zone to deploy to, follow the deployment steps in this article to create a Premium SSD v2 disk and attach it to a VM.

## Use a Premium SSD v2

- [Azure CLI](#)
- [PowerShell](#)
- [Azure portal](#)

Create a Premium SSD v2 disk in an availability zone. Then create a VM in the same region and availability zone that supports Premium Storage and attach the disk to it. Replace the values of all the variables with your own, then run the following script:

```

## Initialize variables
diskName="yourDiskName"
resourceGroupName="yourResourceGroupName"
region="yourRegionName"
zone="yourZoneNumber"
logicalSectorSize=4096
vmName="yourVMName"
vmImage="Win2016Datacenter"
adminPassword="yourAdminPassword"
adminUserName="yourAdminUserName"
vmSize="Standard_D4s_v3"

## Create a Premium SSD v2 disk
az disk create -n $diskName -g $resourceGroupName \
--size-gb 100 \
--disk-iops-read-write 5000 \
--disk-mbps-read-write 150 \
--location $region \
--zone $zone \
--sku PremiumV2_LRS \
--logical-sector-size $logicalSectorSize

## Create the VM
az vm create -n $vmName -g $resourceGroupName \
--image $vmImage \
--zone $zone \
--authentication-type password --admin-password $adminPassword --admin-username $adminUserName \
--size $vmSize \
--location $region \
--attach-data-disks $diskName

```

## Adjust disk performance

Unlike other managed disks, the performance of a Premium SSD v2 can be configured independently of its size. For conceptual information on this, see [Premium SSD v2 performance](#).

- [Azure CLI](#)

- [PowerShell](#)
- [Azure portal](#)

The following command changes the performance of your disk, update the values as you like, then run the command:

```
az disk update `  
--subscription $subscription `  
--resource-group $rgname `  
--name $diskName `  
--set diskIopsReadWrite=5000 `  
--set diskMbpsReadWrite=200
```

## Next steps

# Virtual machine and disk performance

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article helps clarify disk performance and how it works when you combine Azure Virtual Machines and Azure disks. It also describes how you can diagnose bottlenecks for your disk IO and the changes you can make to optimize for performance.

## How does disk performance work?

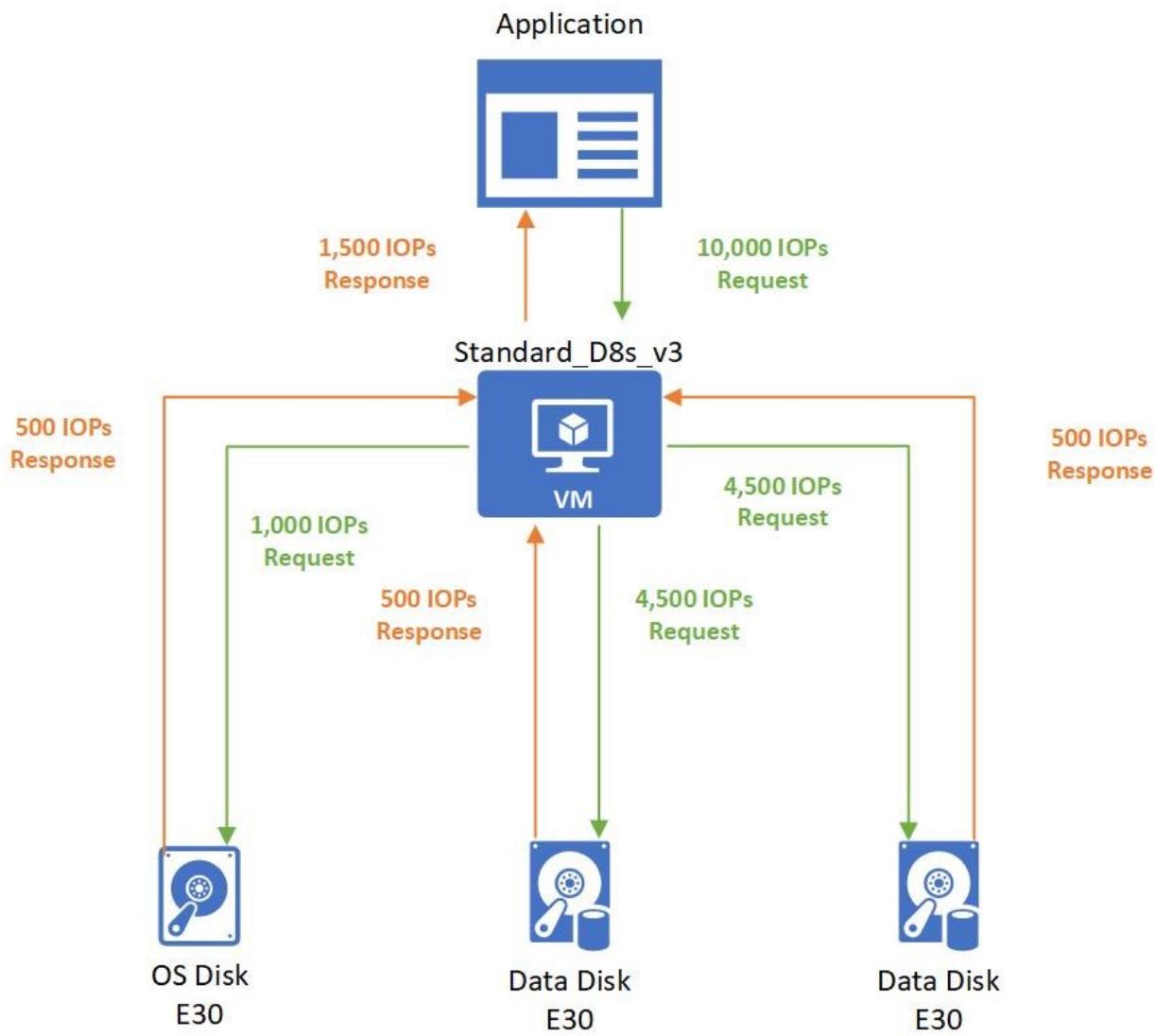
Azure virtual machines have input/output operations per second (IOPS) and throughput performance limits based on the virtual machine type and size. OS disks and data disks can be attached to virtual machines. The disks have their own IOPS and throughput limits.

Your application's performance gets capped when it requests more IOPS or throughput than what is allotted for the virtual machines or attached disks. When capped, the application experiences suboptimal performance. This can lead to negative consequences like increased latency. Let's run through a couple of examples to clarify this concept. To make these examples easy to follow, we'll only look at IOPS. But, the same logic applies to throughput.

## Disk IO capping

Setup:

- Standard\_D8s\_v3
  - Uncached IOPS: 12,800
- E30 OS disk
  - IOPS: 500
- Two E30 data disks × 2
  - IOPS: 500



The application running on the virtual machine makes a request that requires 10,000 IOPS to the virtual machine. All of which are allowed by the VM because the Standard\_D8s\_v3 virtual machine can execute up to 12,800 IOPS.

The 10,000 IOPS requests are broken down into three different requests to the different disks:

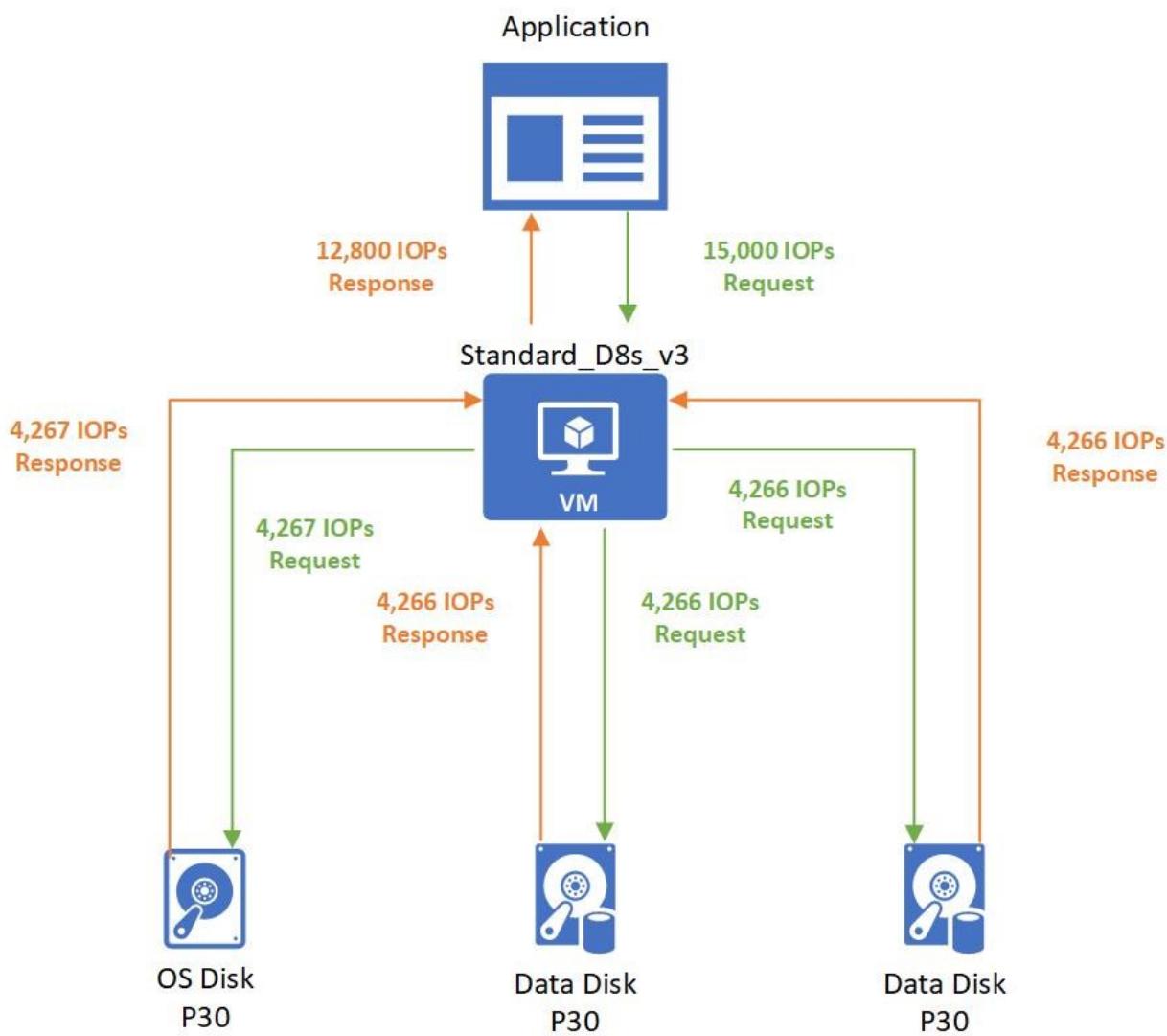
- 1,000 IOPS are requested to the operating system disk.
- 4,500 IOPS are requested to each data disk.

All attached disks are E30 disks and can only handle 500 IOPS. So, they respond back with 500 IOPS each. The application's performance is capped by the attached disks, and it can only process 1,500 IOPS. The application could work at peak performance at 10,000 IOPS if better-performing disks are used, such as Premium SSD P30 disks.

## Virtual machine IO capping

### Setup:

- Standard\_D8s\_v3
  - Uncached IOPS: 12,800
- P30 OS disk
  - IOPS: 5,000
- Two P30 data disks × 2
  - IOPS: 5,000



The application running on the virtual machine makes a request that requires 15,000 IOPS. Unfortunately, the Standard\_D8s\_v3 virtual machine is only provisioned to handle 12,800 IOPS. The application is capped by the virtual machine limits and must allocate the allotted 12,800 IOPS.

Those 12,800 IOPS requested are broken down into three different requests to the different disks:

- 4,267 IOPS are requested to the operating system disk.
- 4,266 IOPS are requested to each data disk.

All attached disks are P30 disks that can handle 5,000 IOPS. So, they respond back with their requested amounts.

## Virtual machine uncached vs cached limits

Virtual machines that are enabled for both premium storage and premium storage caching have two different storage bandwidth limits. Let's look at the Standard\_D8s\_v3 virtual machine as an example. Here is the documentation on the [Dsv3-series](#) and the [Standard\\_D8s\\_v3](#):

Size	vCPU	Memory: GiB	Temp storage (SSD) GiB	Max data disks	<b>Max cached and temp storage throughput: IOPS/MBps</b>	<b>Max uncached disk throughput: IOPS/MBps</b>	<b>Max NICs/Expected network bandwidth (Mbps)</b>
					(cache size in GiB)		
Standard_D2s_v3	2	8	16	4	4000/32 (50)	3200/48	2/1000
Standard_D4s_v3	4	16	32	8	8000/64 (100)	6400/96	2/2000
Standard_D8s_v3	8	32	64	16	16000/128 (200)	12800/192	4/4000
Standard_D16s_v3	16	64	128	32	32000/256 (400)	25600/384	8/8000
Standard_D32s_v3	32	128	256	32	64000/512 (800)	51200/768	8/16000
Standard_D48s_v3	48	192	384	32	96000/768 (1200)	76800/1152	8/24000
Standard_D64s_v3	64	256	512	32	128000/1024 (1600)	80000/1200	8/30000

- The max *uncached* disk throughput is the default storage maximum limit that the virtual machine can handle.
- The max *cached* storage throughput limit is a separate limit when you enable host caching.

Host caching works by bringing storage closer to the VM that can be written or read to quickly. The amount of storage that is available to the VM for host caching is in the documentation. For example, you can see the Standard\_D8s\_v3 comes with 200 GiB of cache storage.

You can enable host caching when you create your virtual machine and attach disks. You can also turn on and off host caching on your disks on an existing VM.

#### OS disk

Swap OS disk						
Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (MBps)	Encryption ⓘ	Host caching ⓘ
throttling-vm-tes	Premium SSD	-	-	-	SSE with PMK	<input checked="" type="checkbox"/> Read/write

#### Data disks

Filter by name

Showing 2 of 2 attached data disks

Create and attach a new disk  Attach existing disks							
LUN ⓘ	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (MBps)	Encryption ⓘ	Host caching ⓘ
0	throttling-vm-test_DataDisk_0	Premium SSD	-	-	-	SSE with PMK	<input checked="" type="checkbox"/> Read-only
1	throttling-vm-test_DataDisk_1	Premium SSD	-	-	-	SSE with PMK	<input checked="" type="checkbox"/> Read/write

You can adjust the host caching to match your workload requirements for each disk. You can set your host caching to be:

- Read-only:** For workloads that only do read operations
- Read/write:** For workloads that do a balance of read and write operations

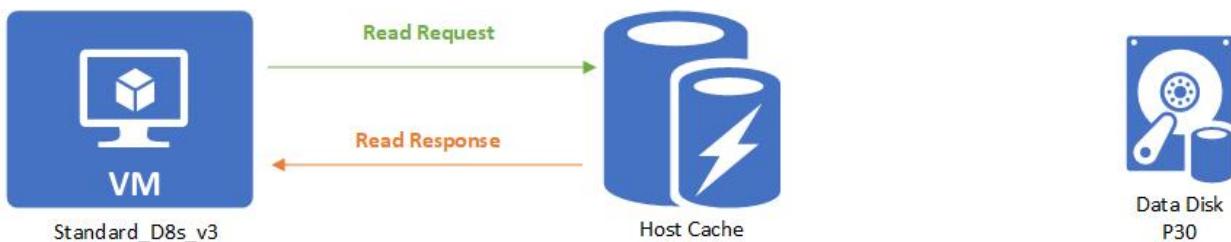
If your workload doesn't follow either of these patterns, we don't recommend that you use host caching.

Let's run through a couple examples of different host cache settings to see how it affects the data flow and performance. In this first example, we'll look at what happens with IO requests when the host caching setting is set to **Read-only**.

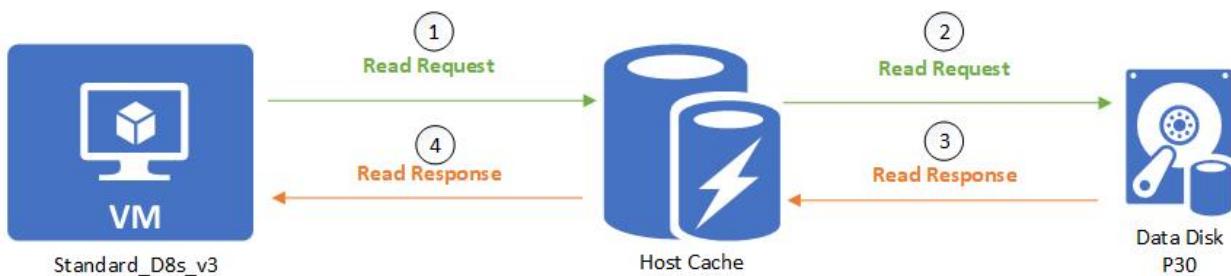
## Setup:

- Standard\_D8s\_v3
  - Cached IOPS: 16,000
  - Uncached IOPS: 12,800
- P30 data disk
  - IOPS: 5,000
  - Host caching: **Read-only**

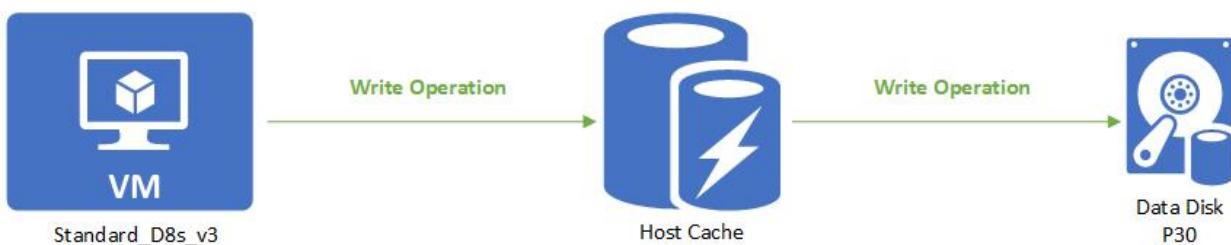
When a read is performed and the desired data is available on the cache, the cache returns the requested data. There is no need to read from the disk. This read is counted toward the VM's cached limits.



When a read is performed and the desired data is *not* available on the cache, the read request is relayed to the disk. Then the disk surfaces it to both the cache and the VM. This read is counted toward both the VM's uncached limit and the VM's cached limit.



When a write is performed, the write has to be written to both the cache and the disk before it is considered complete. This write is counted toward the VM's uncached limit and the VM's cached limit.



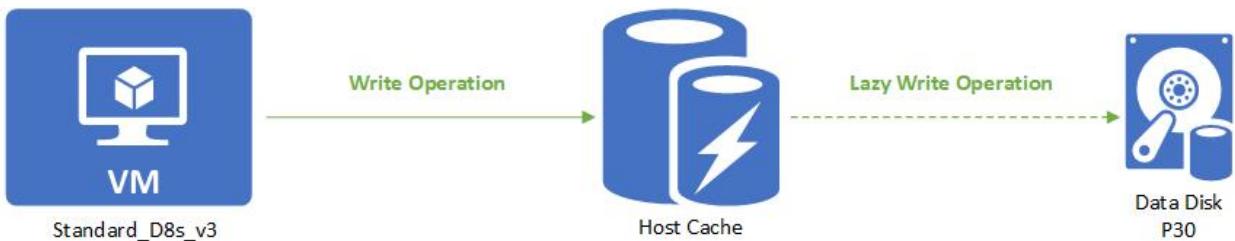
Next let's look at what happens with IO requests when the host cache setting is set to **Read/write**.

## Setup:

- Standard\_D8s\_v3
  - Cached IOPS: 16,000
  - Uncached IOPS: 12,800
- P30 data disk
  - IOPS: 5,000
  - Host caching: **Read/write**

A read is handled the same way as a read-only. Writes are the only thing that's different with read/write caching. When writing with host caching is set to **Read/write**, the write only needs to be written to the host cache to be

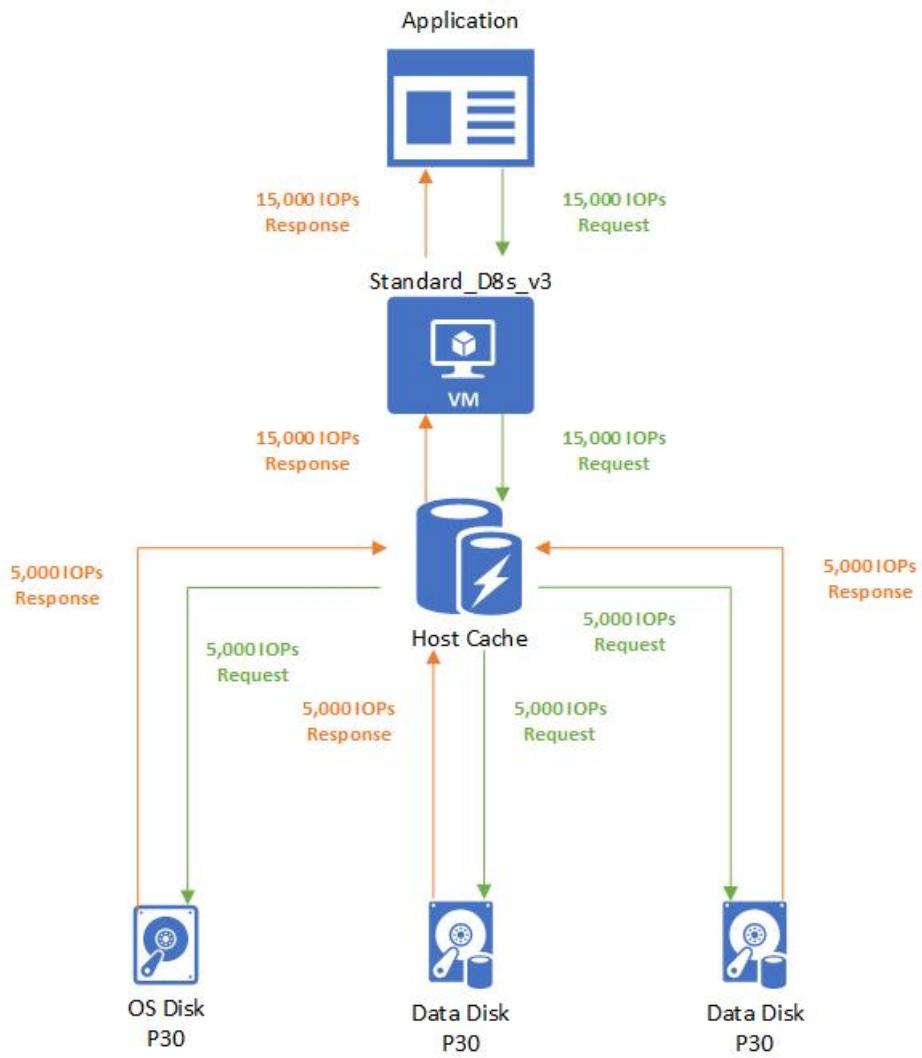
considered complete. The write is then lazily written to the disk as a background process. This means that a write is counted toward cached IO when it is written to the cache. When it is lazily written to the disk, it counts toward the uncached IO.



Let's continue with our Standard\_D8s\_v3 virtual machine. Except this time, we'll enable host caching on the disks. Also, now the VM's IOPS limit is 16,000 IOPS. Attached to the VM are three underlying P30 disks that can each handle 5,000 IOPS.

#### Setup:

- Standard\_D8s\_v3
  - Cached IOPS: 16,000
  - Uncached IOPS: 12,800
- P30 OS disk
  - IOPS: 5,000
  - Host caching: **Read/write**
- Two P30 data disks × 2
  - IOPS: 5,000
  - Host caching: **Read/write**



The application uses a Standard\_D8s\_v3 virtual machine with caching enabled. It makes a request for 15,000 IOPS. The requests are broken down as 5,000 IOPS to each underlying disk attached. No performance capping occurs.

## Combined uncached and cached limits

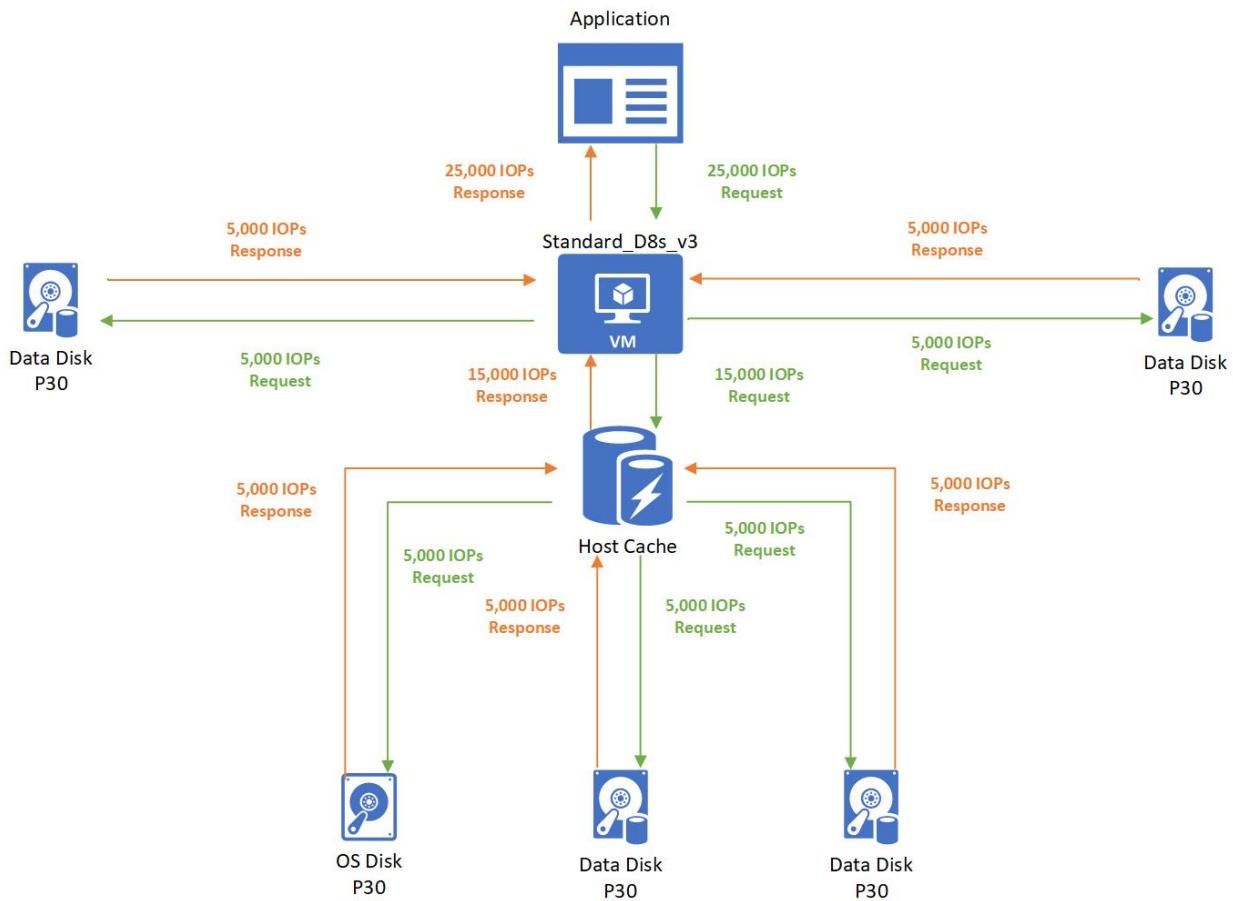
A virtual machine's cached limits are separate from its uncached limits. This means you can enable host caching on disks attached to a VM while not enabling host caching on other disks. This configuration allows your virtual machines to get a total storage IO of the cached limit plus the uncached limit.

Let's run through an example to help you understand how these limits work together. We'll continue with the Standard\_D8s\_v3 virtual machine and premium disks attached configuration.

### Setup:

- Standard\_D8s\_v3
  - Cached IOPS: 16,000
  - Uncached IOPS: 12,800
- P30 OS disk
  - IOPS: 5,000
  - Host caching: **Read/write**
- Two P30 data disks × 2
  - IOPS: 5,000
  - Host caching: **Read/write**
- Two P30 data disks × 2

- IOPS: 5,000
- Host caching: **Disabled**



In this case, the application running on a Standard\_D8s\_v3 virtual machine makes a request for 25,000 IOPS. The request is broken down as 5,000 IOPS to each of the attached disks. Three disks use host caching and two disks don't use host caching.

- Since the three disks that use host caching are within the cached limits of 16,000, those requests are successfully completed. No storage performance capping occurs.
- Since the two disks that don't use host caching are within the uncached limits of 12,800, those requests are also successfully completed. No capping occurs.

# Understand how your reservation discount is applied to Azure disk storage

9/21/2022 • 2 minutes to read • [Edit Online](#)

After you purchase Azure disk reserved capacity, a reservation discount is automatically applied to disk resources that match the terms of your reservation. The reservation discount applies to disk SKUs only. Disk snapshots are charged at pay-as-you-go rates.

For more information about Azure disk reservation, see [Save costs with Azure disk reservation](#). For information about pricing for Azure disk reservation, see [Azure Managed Disks pricing](#).

## How the reservation discount is applied

The Azure disk reservation discount is a use-it-or-lose-it discount. It's applied to managed disk resources hourly. For a given hour, if you have no managed disk resources that meet the reservation terms, you lose a reservation quantity for that hour. Unused hours don't carry forward.

When you delete a resource, the reservation discount automatically applies to another matching resource in the specified scope. If no matching resource is found, the reserved hours are lost.

## Discount examples

The following examples show how the Azure disk reservation discount applies depending on your deployment.

Suppose you purchase and reserve 100 P30 disks in the US West 2 region for a one-year term. Each disk has approximately 1 TiB of storage. Assume the cost of this sample reservation is \$140,100. You can choose to pay either the full amount up front or fixed monthly installments of \$11,675 for the next 12 months.

The following scenarios describe what happens if you underuse, overuse, or tier your reserved capacity. For these examples, assume you've signed up for a monthly reservation-payment plan.

### **Underusing your capacity**

Suppose you deploy only 99 of your 100 reserved Azure premium solid-state drive (SSD) P30 disks for an hour within the reservation period. The remaining P30 disk isn't applied during that hour. It also doesn't carry over.

### **Overusing your capacity**

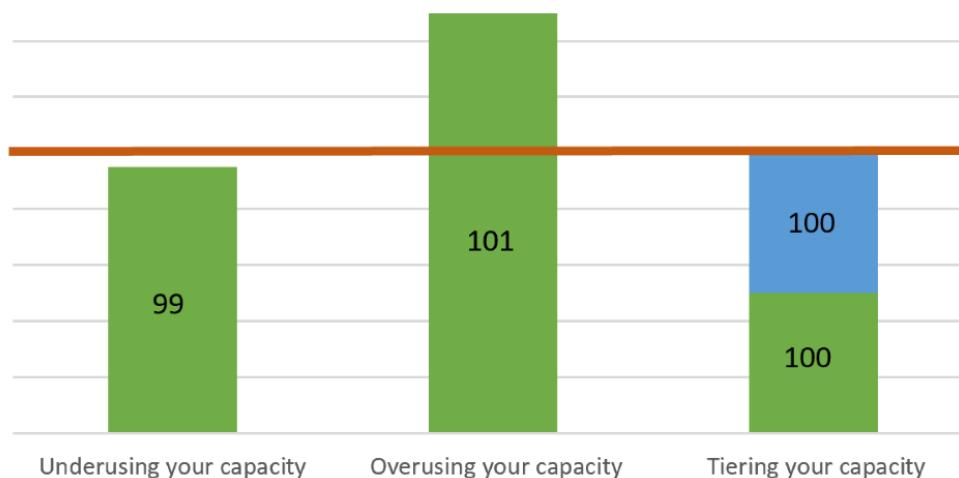
Suppose that for an hour within the reservation period, you use 101 premium SSD P30 disks. The reservation discount applies only to 100 P30 disks. The remaining P30 disk is charged at pay-as-you-go rates for that hour. For the next hour, if your usage goes down to 100 P30 disks, all usage is covered by the reservation.

### **Tiering your capacity**

Suppose that in a given hour within your reservation period, you want to use a total of 200 P30 premium SSDs. Also suppose you use only 100 for the first 30 minutes. During this period, your use is fully covered because you made a reservation for 100 P30 disks. If you then discontinue the use of the first 100 (so that you're using zero) and then begin to use the other 100 for the remaining 30 minutes, that usage is also covered under your reservation.

## Reserved disks example scenarios

One hour



## Need help? Contact us

If you have questions or need help, [create a support request](#).

## Next steps

- [Reduce costs with Azure Disks Reservation](#)
- [What are Azure Reservations?](#)

# Reduce costs with Azure Disks Reservation

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Save on your Azure Disk Storage usage with reserved capacity. Azure Disk Storage reservations combined with Azure Reserved Virtual Machine Instances let you lower your total virtual machine (VM) costs. The reservation discount is applied automatically to the matching disks in the selected reservation scope. Because of this automatic application, you don't need to assign a reservation to a managed disk to get the discounts.

Discounts are applied hourly depending on the disk usage. Unused reserved capacity doesn't carry over. Azure Disk Storage reservation discounts don't apply to unmanaged disks, ultra disks, or page blob consumption.

## Determine your storage needs

Before you purchase a reservation, determine your storage needs. Currently, Azure Disk Storage reservations are available only for select Azure premium SSD SKUs. The SKU of a premium SSD determines the disk's size and performance.

When determining your storage needs, don't think of disks based on just capacity. For example, you can't have a reservation for a P40 disk and use that to pay for two smaller P30 disks. When purchasing a reservation, you're only purchasing a reservation for the total number of disks per SKU.

A disk reservation is made per disk SKU. As a result, the reservation consumption is based on the unit of the disk SKUs instead of the provided size.

For example, assume you reserve one P40 disk that has 2 TiB of provisioned storage capacity. Also assume you allocate only two P30 disks. The P40 reservation in that case doesn't account for P30 consumption, and you pay the pay-as-you-go rate on the P30 disks.

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Dis k size in GiB	4	8	16	32	64	128	256	512	1,0 24	2,0 48	4,0 96	8,1 92	16, 384	32, 767
Pro visi one d IOP S per disk	120	120	120	120	240	500	1,1 00	2,3 00	5,0 00	7,5 00	7,5 00	16, 000	18, 000	20, 000

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Pro vi one d Thr oug hpu t per disk	25 MB /sec	25 MB /sec	25 MB /sec	25 MB /sec	50 MB /sec	100 MB /sec	125 MB /sec	150 MB /sec	200 MB /sec	250 MB/ sec	250 MB/ sec	500 MB/ sec	750 MB/ sec	900 MB/ sec
Ma x bur st IOP S per disk	3,5 00	30, 000 *	30, 000 *	30, 000 *	30, 000 *	30, 000 *	30, 000 *							
Ma x bur st thr oug hpu t per disk	170 MB /sec	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*							
Ma x bur st dur atio n	30 min	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*							
Eligi ble for rese rvat ion	No	Yes, up to one yea r	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year							

\*Applies only to disks with on-demand bursting enabled.

## Purchase considerations

We recommend the following practices when considering disk reservation purchase:

- Analyze your usage information to help determine which reservations you should purchase. Make sure you track the usage in disk SKUs instead of provisioned or used disk capacity.

- Examine your disk reservation along with your VM reservation. We highly recommend making reservations for both VM usage and disk usage for maximum savings. You can start with determining the right VM reservation and then evaluate the disk reservation. Generally, you'll have a standard configuration for each of your workloads. For example, a SQL Server server might have two P40 data disks and one P30 operating system disk.

This kind of pattern can help you determine the reserved amount you might purchase. This approach can simplify the evaluation process and ensure that you have an aligned plan for both your VM and disks. The plan contains considerations like subscriptions or regions.

## Purchase restrictions

Reservation discounts are currently unavailable for the following:

- Unmanaged disks or page blobs.
- Standard SSDs or standard hard-disk drives (HDDs).
- Premium SSD SKUs smaller than P30: P1, P2, P3, P4, P6, P10, P15, and P20 SSD SKUs.
- Disks in Azure Government, Azure Germany, or Azure China regions.

In rare circumstances, Azure limits the purchase of new reservations to a subset of disk SKUs because of low capacity in a region.

## Buy a disk reservation

You can purchase Azure Disk Storage reservations through the [Azure portal](#). You can pay for the reservation either up front or with monthly payments. For more information about purchasing with monthly payments, see [Purchase reservations with monthly payments](#).

Follow these steps to purchase reserved capacity:

- Go to the [Purchase reservations](#) pane in the Azure portal.
- Select **Azure Managed Disks** to purchase a reservation.

The screenshot shows the 'Purchase reservations' pane in the Azure portal. It displays various service options with their descriptions and 'Buy' buttons. The 'Azure Managed Disks' option is highlighted with a red border.

Service	Description	Action
<b>Virtual machine</b> By Microsoft Corp.	Save on virtual machine costs by buying reserved instances for 1 or 3 years	Buy
<b>SQL Database</b> By Microsoft Corp.	Save on SQL Database compute costs by buying reserved vCores for 1 or 3 years	Buy
<b>Azure SQL Data Warehouse</b> By Microsoft Corp.	Save up to 65% on SQL Data Warehouse costs by buying reserved capacity for 1 or 3 years	Buy
<b>Azure Cosmos DB</b> By Microsoft Corp.	Save up to 65% on Cosmos DB by buying reserved throughput capacity for 1 or 3 years	Buy
<b>Azure Blob Storage</b> By Microsoft Corp.	Save on Azure Storage costs for Block Blobs and Azure Data Lake Storage by buying Azure Blob Storage Reserved Capacity for 1 or 3 years	Buy
<b>Azure Database for MySQL</b> By Microsoft Corp.	Save on Azure Database for MySQL compute costs by buying reserved vCores for 1 year	Buy
<b>Azure Database for MariaDB</b> By Microsoft Corp.	Save on Azure Database for MariaDB compute costs by buying reserved vCores for 1 year	Buy
<b>Azure Database for PostgreSQL</b> By Microsoft Corp.	Save on Azure Database for PostgreSQL single server compute costs by buying reserved vCores for 1 year	Buy
<b>Azure Managed Disks</b> By Microsoft Corp.	Save on Premium SSD Managed Disks by buying reserved disks for 1 year	Buy
<b>Azure Databricks</b> By Microsoft Corp.	Save on your Azure Databricks costs by pre-purchasing DBUs for 1 or 3 years	Buy

**Next: Review + buy**

- Specify the required values described in the following table:

ELEMENT	DESCRIPTION
<b>Scope</b>	<p>How many subscriptions can use the billing benefit associated with the reservation. This value also specifies how the reservation is applied to specific subscriptions.</p> <p>If you select <b>Shared</b>, the reservation discount is applied to Azure Storage capacity in every subscription within your billing context. The billing context is based on how you signed up for Azure. For enterprise customers, the shared scope is the enrollment and includes all subscriptions within the enrollment. For pay-as-you-go customers, the shared scope includes all individual subscriptions with pay-as-you-go rates created by the account administrator.</p> <p>If you select <b>Management group</b>, the reservation discount is applied to Azure Storage capacity in every subscription that is part of the management group and the billing scope.</p> <p>If you select <b>Single subscription</b>, the reservation discount is applied to Azure Storage capacity in the selected subscription.</p> <p>If you select <b>Single resource group</b>, the reservation discount is applied to Azure Storage capacity in the selected subscription and in that subscription's selected resource group.</p> <p>You can change the reservation scope after you purchase the reservation.</p>
<b>Subscription</b>	<p>The subscription you use to pay for the Azure Storage reservation. The payment method on the selected subscription is used in charging the costs. The subscription must be one of the following types:</p> <ul style="list-style-type: none"> <li>• Enterprise Agreement (offer numbers MS-AZR-0017P and MS-AZR-0148P). For an Enterprise subscription, the charges are deducted from the enrollment's Azure Prepayment (previously called monetary commitment) balance or charged as overage.</li> <li>• Individual subscription with pay-as-you-go rates (offer numbers MS-AZR-0003P and MS-AZR-0023P). For an individual subscription with pay-as-you-go rates, the charges are billed to the credit card or invoice payment method on the subscription.</li> </ul>
<b>Disks</b>	The SKU you want to create.
<b>Region</b>	The region where the reservation is in effect.
<b>Billing frequency</b>	How often the account is billed for the reservation. Options include <b>Monthly</b> and <b>Upfront</b> .

Select the product you want to purchase

Save on your Premium SSD Managed Disks usage by purchasing reserved capacity. Discounts are applied hourly on the disk usage, any unused reserved capacity does not carry over. Reservation discount does not apply to Premium SSD Unmanaged Disks or Page Blob consumption. [Learn More](#)

Scope \* Shared Subscription \* Sub for RI Testing

Filter by name... Region : West Europe Disk : Select a value Billing frequency : Select a value Reset filters

Name	Disk	Region	Term	Billing frequency
Premium SSD Managed Disks	P30	West Europe	One Year	Upfront
Premium SSD Managed Disks	P30	West Europe	One Year	Monthly
Premium SSD Managed Disks	P40	West Europe	One Year	Upfront
Premium SSD Managed Disks	P40	West Europe	One Year	Monthly
Premium SSD Managed Disks	P50	West Europe	One Year	Upfront
Premium SSD Managed Disks	P50	West Europe	One Year	Monthly
Premium SSD Managed Disks	P60	West Europe	One Year	Upfront
Premium SSD Managed Disks	P60	West Europe	One Year	Monthly
Premium SSD Managed Disks	P70	West Europe	One Year	Upfront
Premium SSD Managed Disks	P70	West Europe	One Year	Monthly
Premium SSD Managed Disks	P80	West Europe	One Year	Upfront
Premium SSD Managed Disks	P80	West Europe	One Year	Monthly

Select Cancel

- After you specify the values for your reservation, the Azure portal displays the cost. The portal also shows the discount percentage over pay-as-you-go billing. Select **Next** to continue to the **Purchase reservations** pane.
- On the **Purchase reservations** pane, you can name your reservation and select the total quantity of reservations you want to make. The number of reservations maps to the number of disks. For example, if you want to reserve a hundred disks, enter the **Quantity** value 100.
- Review the total cost of the reservation.

Home > Reservations > Purchase reservations

Purchase reservations

Products Review + buy

Azure Managed Disks

Download Cart

Reservation name	Product	Scope	Unit price	Quantity	Subtotal (% Discount)	Billing frequency
Disk_RI_01-14-2020_10-31	Premium SSD Managed Disks   P30   West Europe   One Year	Shared	<price>	1	<subtotal>	Upfront

Next: Review + buy Total reservation cost <total-cost>

After you purchase a reservation, it's automatically applied to any existing Disk Storage resources that match the reservation terms. If you haven't created any Disk Storage resources yet, the reservation applies whenever you create a resource that matches the reservation terms. In either case, the reservation term begins immediately after a successful purchase.

## Cancel, exchange, or refund reservations

You can cancel, exchange, or refund reservations within certain limitations. For more information, see [Self-service exchanges and refunds for Azure Reservations](#).

## Expiration of a reservation

When a reservation expires, any Azure Disk Storage capacity that you use under that reservation is billed at the pay-as-you-go rate. Reservations don't renew automatically.

You'll receive an email notification 30 days before the expiration of the reservation and again on the expiration date. To continue taking advantage of the cost savings that a reservation provides, renew it no later than the expiration date.

## Need help? Contact us

If you have questions or need help, [create a support request](#).

## Next steps

- [What are Azure Reservations?](#)
- [Understand how your reservation discount is applied to Azure Disk Storage](#)

# Azure premium storage: design for high performance

9/21/2022 • 36 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article provides guidelines for building high performance applications using Azure Premium Storage. You can use the instructions provided in this document combined with performance best practices applicable to technologies used by your application. To illustrate the guidelines, we have used SQL Server running on Premium Storage as an example throughout this document.

While we address performance scenarios for the Storage layer in this article, you will need to optimize the application layer. For example, if you are hosting a SharePoint Farm on Azure Premium Storage, you can use the SQL Server examples from this article to optimize the database server. Additionally, optimize the SharePoint Farm's Web server and Application server to get the most performance.

This article will help answer following common questions about optimizing application performance on Azure Premium Storage,

- How to measure your application performance?
- Why are you not seeing expected high performance?
- Which factors influence your application performance on Premium Storage?
- How do these factors influence performance of your application on Premium Storage?
- How can you optimize for IOPS, Bandwidth and Latency?

We have provided these guidelines specifically for Premium Storage because workloads running on Premium Storage are highly performance sensitive. We have provided examples where appropriate. You can also apply some of these guidelines to applications running on IaaS VMs with Standard Storage disks.

## NOTE

Sometimes, what appears to be a disk performance issue is actually a network bottleneck. In these situations, you should optimize your [network performance](#).

If you are looking to benchmark your disk, see our articles on benchmarking a disk:

- For Linux: [Benchmark your application on Azure Disk Storage](#)
- For Windows: [Benchmarking a disk](#).

If your VM supports accelerated networking, you should make sure it is enabled. If it is not enabled, you can enable it on already deployed VMs on both [Windows](#) and [Linux](#).

Before you begin, if you are new to Premium Storage, first read the [Select an Azure disk type for IaaS VMs](#) and [Scalability targets for premium page blob storage accounts](#).

## Application performance indicators

We assess whether an application is performing well or not using performance indicators like, how fast an application is processing a user request, how much data an application is processing per request, how many requests an application is processing in a specific period of time, how long a user has to wait to get a response after submitting their request. The technical terms for these performance indicators are, IOPS, Throughput or

Bandwidth, and Latency.

In this section, we will discuss the common performance indicators in the context of Premium Storage. In the following section, Gathering Application Requirements, you will learn how to measure these performance indicators for your application. Later in [Optimize application performance](#), you will learn about the factors affecting these performance indicators and recommendations to optimize them.

## IOPS

IOPS, or Input/output Operations Per Second, is the number of requests that your application is sending to the storage disks in one second. An input/output operation could be read or write, sequential, or random. Online Transaction Processing (OLTP) applications like an online retail website need to process many concurrent user requests immediately. The user requests are insert and update intensive database transactions, which the application must process quickly. Therefore, OLTP applications require very high IOPS. Such applications handle millions of small and random IO requests. If you have such an application, you must design the application infrastructure to optimize for IOPS. In [Optimize application performance](#), we discuss in detail all the factors that you must consider to get high IOPS.

When you attach a premium storage disk to your high scale VM, Azure provisions for you a guaranteed number of IOPS as per the disk specification. For example, a P50 disk provisions 7500 IOPS. Each high scale VM size also has a specific IOPS limit that it can sustain. For example, a Standard GS5 VM has 80,000 IOPS limit.

## Throughput

Throughput, or bandwidth is the amount of data that your application is sending to the storage disks in a specified interval. If your application is performing input/output operations with large IO unit sizes, it requires high throughput. Data warehouse applications tend to issue scan intensive operations that access large portions of data at a time and commonly perform bulk operations. In other words, such applications require higher throughput. If you have such an application, you must design its infrastructure to optimize for throughput. In the next section, we discuss in detail the factors you must tune to achieve this.

When you attach a premium storage disk to a high scale VM, Azure provisions throughput as per that disk specification. For example, a P50 disk provisions 250 MB per second disk throughput. Each high scale VM size also has a specific throughput limit that it can sustain. For example, Standard GS5 VM has a maximum throughput of 2,000 MB per second.

There is a relation between throughput and IOPS as shown in the formula below.

$$\text{IOPS} \times \text{IO Size} = \text{Throughput}$$

Therefore, it is important to determine the optimal throughput and IOPS values that your application requires. As you try to optimize one, the other also gets affected. In [Optimize application performance](#), we will discuss in more details about optimizing IOPS and Throughput.

## Latency

Latency is the time it takes an application to receive a single request, send it to the storage disks and send the response to the client. This is a critical measure of an application's performance in addition to IOPS and Throughput. The Latency of a premium storage disk is the time it takes to retrieve the information for a request and communicate it back to your application. Premium Storage provides consistent low latencies. Premium Disks are designed to provide single-digit millisecond latencies for most IO operations. If you enable `ReadOnly` host caching on premium storage disks, you can get much lower read latency. We discuss Disk Caching in more detail in [Disk caching](#).

When you are optimizing your application to get higher IOPS and Throughput, it will affect the latency of your application. After tuning the application performance, always evaluate the latency of the application to avoid unexpected high latency behavior.

The following control plane operations on Managed Disks may involve movement of the Disk from one Storage location to another. This is orchestrated via background copy of data that can take several hours to complete, typically less than 24 hours depending on the amount of data in the disks. During that time your application can experience higher than usual read latency as some reads can get redirected to the original location and can take longer to complete. There is no impact on write latency during this period.

- Update the storage type.
- Detach and attach a disk from one VM to another.
- Create a managed disk from a VHD.
- Create a managed disk from a snapshot.
- Convert unmanaged disks to managed disks.

## Performance Application Checklist for disks

The first step in designing high-performance applications running on Azure Premium Storage is understanding the performance requirements of your application. After you have gathered performance requirements, you can optimize your application to achieve the most optimal performance.

In the previous section, we explained the common performance indicators, IOPS, Throughput, and Latency. You must identify which of these performance indicators are critical to your application to deliver the desired user experience. For example, high IOPS matters most to OLTP applications processing millions of transactions in a second. Whereas, high Throughput is critical for Data Warehouse applications processing large amounts of data in a second. Extremely low Latency is crucial for real-time applications like live video streaming websites.

Next, measure the maximum performance requirements of your application throughout its lifetime. Use the sample checklist below as a start. Record the maximum performance requirements during normal, peak, and off-hours workload periods. By identifying requirements for all workloads levels, you will be able to determine the overall performance requirement of your application. For example, the normal workload of an e-commerce website will be the transactions it serves during most days in a year. The peak workload of the website will be the transactions it serves during holiday season or special sale events. The peak workload is typically experienced for a limited period, but can require your application to scale two or more times its normal operation. Find out the 50 percentile, 90 percentile, and 99 percentile requirements. This helps filter out any outliers in the performance requirements and you can focus your efforts on optimizing for the right values.

## Application performance requirements checklist

PERFORMANCE REQUIREMENTS	50 PERCENTILE	90 PERCENTILE	99 PERCENTILE
Max. Transactions per second			
% Read operations			
% Write operations			
% Random operations			
% Sequential operations			

PERFORMANCE REQUIREMENTS	50 PERCENTILE	90 PERCENTILE	99 PERCENTILE
IO request size			
Average Throughput			
Max. Throughput			
Min. Latency			
Average Latency			
Max. CPU			
Average CPU			
Max. Memory			
Average Memory			
Queue Depth			

#### NOTE

You should consider scaling these numbers based on expected future growth of your application. It is a good idea to plan for growth ahead of time, because it could be harder to change the infrastructure for improving performance later.

If you have an existing application and want to move to Premium Storage, first build the checklist above for the existing application. Then, build a prototype of your application on Premium Storage and design the application based on guidelines described in [Optimize application performance](#). The next article describes the tools you can use to gather the performance measurements.

#### Counters to measure application performance requirements

The best way to measure performance requirements of your application, is to use performance-monitoring tools provided by the operating system of the server. You can use PerfMon for Windows and iostat for Linux. These tools capture counters corresponding to each measure explained in the above section. You must capture the values of these counters when your application is running its normal, peak, and off-hours workloads.

The PerfMon counters are available for processor, memory and, each logical disk and physical disk of your server. When you use premium storage disks with a VM, the physical disk counters are for each premium storage disk, and logical disk counters are for each volume created on the premium storage disks. You must capture the values for the disks that host your application workload. If there is a one to one mapping between logical and physical disks, you can refer to physical disk counters; otherwise refer to the logical disk counters. On Linux, the iostat command generates a CPU and disk utilization report. The disk utilization report provides statistics per physical device or partition. If you have a database server with its data and logs on separate disks, collect this data for both disks. Below table describes counters for disks, processors, and memory:

COUNTER	DESCRIPTION	PERFMON	IOSTAT
IOPS or Transactions per second	Number of I/O requests issued to the storage disk per second.	Disk Reads/sec Disk Writes/sec	tps r/s w/s

COUNTER	DESCRIPTION	PERFMON	IOSTAT
Disk Reads and Writes	% of Reads and Write operations performed on the disk.	% Disk Read Time % Disk Write Time	r/s w/s
Throughput	Amount of data read from or written to the disk per second.	Disk Read Bytes/sec Disk Write Bytes/sec	kB_read/s kB_wrtn/s
Latency	Total time to complete a disk IO request.	Average Disk sec/Read Average disk sec/Write	await svctm
IO size	The size of I/O requests issued to the storage disks.	Average Disk Bytes/Read Average Disk Bytes/Write	avgrq-sz
Queue Depth	Number of outstanding I/O requests waiting to be read from or written to the storage disk.	Current Disk Queue Length	avgqu-sz
Max. Memory	Amount of memory required to run application smoothly	% Committed Bytes in Use	Use vmstat
Max. CPU	Amount CPU required to run application smoothly	% Processor time	%util

Learn more about [iostat](#) and [PerfMon](#).

## Optimize application performance

The main factors that influence performance of an application running on Premium Storage are Nature of IO requests, VM size, Disk size, Number of disks, disk caching, multithreading, and queue depth. You can control some of these factors with knobs provided by the system. Most applications may not give you an option to alter the IO size and Queue Depth directly. For example, if you are using SQL Server, you cannot choose the IO size and queue depth. SQL Server chooses the optimal IO size and queue depth values to get the most performance. It is important to understand the effects of both types of factors on your application performance, so that you can provision appropriate resources to meet performance needs.

Throughout this section, refer to the application requirements checklist that you created, to identify how much you need to optimize your application performance. Based on that, you will be able to determine which factors from this section you will need to tune. To witness the effects of each factor on your application performance, run benchmarking tools on your application setup. Refer to the Benchmarking article, linked at the end, for steps to run common benchmarking tools on Windows and Linux VMs.

### Optimize IOPS, throughput, and latency at a glance

The table below summarizes performance factors and the steps necessary to optimize IOPS, throughput, and latency. The sections following this summary will describe each factor in much more depth.

For more information on VM sizes and on the IOPS, throughput, and latency available for each type of VM, see [Sizes for virtual machines in Azure](#).

	IOPS	Throughput	Latency
<b>Example Scenario</b>	Enterprise OLTP application requiring very high transactions per second rate.	Enterprise Data warehousing application processing large amounts of data.	Near real-time applications requiring instant responses to user requests, like online gaming.
<b>Performance factors</b>			
<b>IO size</b>	Smaller IO size yields higher IOPS.	Larger IO size to yields higher Throughput.	
<b>VM size</b>	Use a VM size that offers IOPS greater than your application requirement.	Use a VM size with throughput limit greater than your application requirement.	Use a VM size that offers scale limits greater than your application requirement.
<b>Disk size</b>	Use a disk size that offers IOPS greater than your application requirement.	Use a disk size with Throughput limit greater than your application requirement.	Use a disk size that offers scale limits greater than your application requirement.
<b>VM and Disk Scale Limits</b>	IOPS limit of the VM size chosen should be greater than total IOPS driven by storage disks attached to it.	Throughput limit of the VM size chosen should be greater than total Throughput driven by premium storage disks attached to it.	Scale limits of the VM size chosen must be greater than total scale limits of attached premium storage disks.
<b>Disk Caching</b>	Enable ReadOnly Cache on premium storage disks with Read heavy operations to get higher Read IOPS.		Enable ReadOnly Cache on premium storage disks with Read heavy operations to get very low Read latencies.
<b>Disk Striping</b>	Use multiple disks and stripe them together to get a combined higher IOPS and Throughput limit. The combined limit per VM should be higher than the combined limits of attached premium disks.		
<b>Stripe Size</b>	Smaller stripe size for random small IO pattern seen in OLTP applications. For example, use stripe size of 64 KB for SQL Server OLTP application.	Larger stripe size for sequential large IO pattern seen in Data Warehouse applications. For example, use 256 KB stripe size for SQL Server Data warehouse application.	
<b>Multithreading</b>	Use multithreading to push higher number of requests to Premium Storage that will lead to higher IOPS and Throughput. For example, on SQL Server set a high MAXDOP value to allocate more CPUs to SQL Server.		

	IOPS	THROUGHPUT	LATENCY
Queue Depth	Larger Queue Depth yields higher IOPS.	Larger Queue Depth yields higher Throughput.	Smaller Queue Depth yields lower latencies.

## Nature of IO requests

An IO request is a unit of input/output operation that your application will be performing. Identifying the nature of IO requests, random or sequential, read or write, small or large, will help you determine the performance requirements of your application. It is important to understand the nature of IO requests, to make the right decisions when designing your application infrastructure. IOs must be distributed evenly to achieve the best performance possible.

IO size is one of the more important factors. The IO size is the size of the input/output operation request generated by your application. The IO size has a significant impact on performance especially on the IOPS and Bandwidth that the application is able to achieve. The following formula shows the relationship between IOPS, IO size, and Bandwidth/Throughput.

$$\text{IOPS} \times \text{IO Size} = \text{Throughput}$$

Some applications allow you to alter their IO size, while some applications do not. For example, SQL Server determines the optimal IO size itself, and does not provide users with any knobs to change it. On the other hand, Oracle provides a parameter called [DB\\_BLOCK\\_SIZE](#) using which you can configure the I/O request size of the database.

If you are using an application, which does not allow you to change the IO size, use the guidelines in this article to optimize the performance KPI that is most relevant to your application. For example,

- An OLTP application generates millions of small and random IO requests. To handle these types of IO requests, you must design your application infrastructure to get higher IOPS.
- A data warehousing application generates large and sequential IO requests. To handle these types of IO requests, you must design your application infrastructure to get higher Bandwidth or Throughput.

If you are using an application, which allows you to change the IO size, use this rule of thumb for the IO size in addition to other performance guidelines,

- Smaller IO size to get higher IOPS. For example, 8 KB for an OLTP application.
- Larger IO size to get higher Bandwidth/Throughput. For example, 1024 KB for a data warehouse application.

Here is an example on how you can calculate the IOPS and Throughput/Bandwidth for your application.

Consider an application using a P30 disk. The maximum IOPS and Throughput/Bandwidth a P30 disk can achieve is 5000 IOPS and 200 MB per second respectively. Now, if your application requires the maximum IOPS from the P30 disk and you use a smaller IO size like 8 KB, the resulting Bandwidth you will be able to get is 40 MB per second. However, if your application requires the maximum Throughput/Bandwidth from P30 disk, and you use a larger IO size like 1024 KB, the resulting IOPS will be less, 200 IOPS. Therefore, tune the IO size such that it meets both your application's IOPS and Throughput/Bandwidth requirement. The following table summarizes the different IO sizes and their corresponding IOPS and Throughput for a P30 disk.

APPLICATION REQUIREMENT	I/O SIZE	IOPS	THROUGHPUT/BANDWIDTH
Max IOPS	8 KB	5,000	40 MB per second

APPLICATION REQUIREMENT	I/O SIZE	IOPS	THROUGHPUT/BANDWIDTH
Max Throughput	1024 KB	200	200 MB per second
Max Throughput + high IOPS	64 KB	3,200	200 MB per second
Max IOPS + high Throughput	32 KB	5,000	160 MB per second

To get IOPS and Bandwidth higher than the maximum value of a single premium storage disk, use multiple premium disks striped together. For example, stripe two P30 disks to get a combined IOPS of 10,000 IOPS or a combined Throughput of 400 MB per second. As explained in the next section, you must use a VM size that supports the combined disk IOPS and Throughput.

#### NOTE

As you increase either IOPS or Throughput the other also increases, make sure you do not hit throughput or IOPS limits of the disk or VM when increasing either one.

To witness the effects of IO size on application performance, you can run benchmarking tools on your VM and disks. Create multiple test runs and use different IO size for each run to see the impact. Refer to the Benchmarking article, linked at the end, for more details.

## High scale VM sizes

When you start designing an application, one of the first things to do is, choose a VM to host your application. Premium Storage comes with High Scale VM sizes that can run applications requiring higher compute power and a high local disk I/O performance. These VMs provide faster processors, a higher memory-to-core ratio, and a Solid-State Drive (SSD) for the local disk. Examples of High Scale VMs supporting Premium Storage are the DS and GS series VMs.

High Scale VMs are available in different sizes with a different number of CPU cores, memory, OS, and temporary disk size. Each VM size also has maximum number of data disks that you can attach to the VM. Therefore, the chosen VM size will affect how much processing, memory, and storage capacity is available for your application. It also affects the Compute and Storage cost. For example, below are the specifications of the largest VM size in a DS series and a GS series:

VM SIZE	CPU CORES	MEMORY	VM DISK SIZES	MAX. DATA DISKS	CACHE SIZE	IOPS	BANDWIDTH CACHE IO LIMITS
Standard_DS14	16	112 GB	OS = 1023 GB Local SSD = 224 GB	32	576 GB	50,000 IOPS 512 MB per second	4,000 IOPS and 33 MB per second
Standard_GS5	32	448 GB	OS = 1023 GB Local SSD = 896 GB	64	4224 GB	80,000 IOPS 2,000 MB per second	5,000 IOPS and 50 MB per second

To view a complete list of all available Azure VM sizes, refer to [Sizes for virtual machines in Azure](#). Choose a VM size that can meet and scale to your desired application performance requirements. In addition to this, take into

account following important considerations when choosing VM sizes.

#### *Scale Limits*

The maximum IOPS limits per VM and per disk are different and independent of each other. Make sure that the application is driving IOPS within the limits of the VM as well as the premium disks attached to it. Otherwise, application performance will experience throttling.

As an example, suppose an application requirement is a maximum of 4,000 IOPS. To achieve this, you provision a P30 disk on a DS1 VM. The P30 disk can deliver up to 5,000 IOPS. However, the DS1 VM is limited to 3,200 IOPS. Consequently, the application performance will be constrained by the VM limit at 3,200 IOPS and there will be degraded performance. To prevent this situation, choose a VM and disk size that will both meet application requirements.

#### *Cost of Operation*

In many cases, it is possible that your overall cost of operation using Premium Storage is lower than using Standard Storage.

For example, consider an application requiring 16,000 IOPS. To achieve this performance, you will need a Standard\_D14 Azure IaaS VM, which can give a maximum IOPS of 16,000 using 32 standard storage 1 TB disks. Each 1-TB standard storage disk can achieve a maximum of 500 IOPS. The estimated cost of this VM per month will be \$1,570. The monthly cost of 32 standard storage disks will be \$1,638. The estimated total monthly cost will be \$3,208.

However, if you hosted the same application on Premium Storage, you will need a smaller VM size and fewer premium storage disks, thus reducing the overall cost. A Standard\_DS13 VM can meet the 16,000 IOPS requirement using four P30 disks. The DS13 VM has a maximum IOPS of 25,600 and each P30 disk has a maximum IOPS of 5,000. Overall, this configuration can achieve  $5,000 \times 4 = 20,000$  IOPS. The estimated cost of this VM per month will be \$1,003. The monthly cost of four P30 premium storage disks will be \$544.34. The estimated total monthly cost will be \$1,544.

Table below summarizes the cost breakdown of this scenario for Standard and Premium Storage.

	STANDARD	PREMIUM
Cost of VM per month	\$1,570.58 (Standard_D14)	\$1,003.66 (Standard_DS13)
Cost of Disks per month	\$1,638.40 (32 x 1-TB disks)	\$544.34 (4 x P30 disks)
Overall Cost per month	\$3,208.98	\$1,544.34

#### *Linux Distros*

With Azure Premium Storage, you get the same level of Performance for VMs running Windows and Linux. We support many flavors of Linux distros. For more information, see [Linux distributions endorsed on Azure](#). It is important to note that different distros are better suited for different types of workloads. You will see different levels of performance depending on the distro your workload is running on. Test the Linux distros with your application and choose the one that works best.

When running Linux with Premium Storage, check the latest updates about required drivers to ensure high performance.

## Premium storage disk sizes

Azure Premium Storage offers a variety of sizes so you can choose one that best suits your needs. Each disk size has a different scale limit for IOPS, bandwidth, and storage. Choose the right Premium Storage Disk size depending on the application requirements and the high scale VM size. The table below shows the disks sizes

and their capabilities. P4, P6, P15, P60, P70, and P80 sizes are currently only supported for Managed Disks.

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Eligi ble for rese rvat ion	No	No	No	No	No	No	No	No	Yes, up to one yea r	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year

\*Applies only to disks with on-demand bursting enabled.

How many disks you choose depends on the disk size chosen. You could use a single P50 disk or multiple P10 disks to meet your application requirement. Take into account considerations listed below when making the choice.

#### *Scale Limits (IOPS and Throughput)*

The IOPS and Throughput limits of each Premium disk size is different and independent from the VM scale limits. Make sure that the total IOPS and Throughput from the disks are within scale limits of the chosen VM size.

For example, if an application requirement is a maximum of 250 MB/sec Throughput and you are using a DS4 VM with a single P30 disk. The DS4 VM can give up to 256 MB/sec Throughput. However, a single P30 disk has Throughput limit of 200 MB/sec. Consequently, the application will be constrained at 200 MB/sec due to the disk limit. To overcome this limit, provision more than one data disks to the VM or resize your disks to P40 or P50.

#### **NOTE**

Reads served by the cache are not included in the disk IOPS and Throughput, hence not subject to disk limits. Cache has its separate IOPS and Throughput limit per VM.

For example, initially your reads and writes are 60MB/sec and 40MB/sec respectively. Over time, the cache warms up and serves more and more of the reads from the cache. Then, you can get higher write Throughput from the disk.

#### *Number of Disks*

Determine the number of disks you will need by assessing application requirements. Each VM size also has a limit on the number of disks that you can attach to the VM. Typically, this is twice the number of cores. Ensure that the VM size you choose can support the number of disks needed.

Remember, the Premium Storage disks have higher performance capabilities compared to Standard Storage disks. Therefore, if you are migrating your application from Azure IaaS VM using Standard Storage to Premium Storage, you will likely need fewer premium disks to achieve the same or higher performance for your application.

## Disk caching

High Scale VMs that leverage Azure Premium Storage have a multi-tier caching technology called BlobCache. BlobCache uses a combination of the host RAM and local SSD for caching. This cache is available for the Premium Storage persistent disks and the VM local disks. By default, this cache setting is set to Read/Write for OS disks and ReadOnly for data disks hosted on Premium Storage. With disk caching enabled on the Premium Storage disks, the high scale VMs can achieve extremely high levels of performance that exceed the underlying disk performance.

### WARNING

Disk Caching is not supported for disks 4 TiB and larger. If multiple disks are attached to your VM, each disk that is smaller than 4 TiB will support caching.

Changing the cache setting of an Azure disk detaches and re-attaches the target disk. If it is the operating system disk, the VM is restarted. Stop all applications/services that might be affected by this disruption before changing the disk cache setting. Not following those recommendations could lead to data corruption.

To learn more about how BlobCache works, refer to the [Inside Azure Premium Storage](#) blog post.

It is important to enable cache on the right set of disks. Whether you should enable disk caching on a premium disk or not will depend on the workload pattern that disk will be handling. Table below shows the default cache settings for OS and Data disks.

DISK TYPE	DEFAULT CACHE SETTING
OS disk	ReadWrite
Data disk	ReadOnly

Following are the recommended disk cache settings for data disks,

DISK CACHING SETTING	RECOMMENDATION ON WHEN TO USE THIS SETTING
None	Configure host-cache as None for write-only and write-heavy disks.
ReadOnly	Configure host-cache as ReadOnly for read-only and read-write disks.
ReadWrite	Configure host-cache as ReadWrite only if your application properly handles writing cached data to persistent disks when needed.

### *ReadOnly*

By configuring ReadOnly caching on Premium Storage data disks, you can achieve low Read latency and get very high Read IOPS and Throughput for your application. This is due two reasons,

1. Reads performed from cache, which is on the VM memory and local SSD, are much faster than reads from the data disk, which is on the Azure blob storage.
2. Premium Storage does not count the Reads served from cache, towards the disk IOPS and Throughput. Therefore, your application is able to achieve higher total IOPS and Throughput.

### *ReadWrite*

By default, the OS disks have ReadWrite caching enabled. We have recently added support for ReadWrite caching on data disks as well. If you are using ReadWrite caching, you must have a proper way to write the data from cache to persistent disks. For example, SQL Server handles writing cached data to the persistent storage disks on its own. Using ReadWrite cache with an application that does not handle persisting the required data can lead to data loss, if the VM crashes.

### *None*

Currently, **None** is only supported on data disks. It is not supported on OS disks. If you set **None** on an OS disk it will override this internally and set it to **ReadOnly**.

As an example, you can apply these guidelines to SQL Server running on Premium Storage by doing the

following,

1. Configure "ReadOnly" cache on premium storage disks hosting data files.
  - a. The fast reads from cache lower the SQL Server query time since data pages are retrieved much faster from the cache compared to directly from the data disks.
  - b. Serving reads from cache, means there is additional Throughput available from premium data disks. SQL Server can use this additional Throughput towards retrieving more data pages and other operations like backup/restore, batch loads, and index rebuilds.
2. Configure "None" cache on premium storage disks hosting the log files.
  - a. Log files have primarily write-heavy operations. Therefore, they do not benefit from the ReadOnly cache.

## Optimize performance on Linux VMs

For all premium SSDs or ultra disks, you may be able to disable "barriers" for file systems on the disk in order to improve performance when it is known that there are no caches that could lose data. If Azure disk caching is set to ReadOnly or None, you can disable barriers. But if caching is set to ReadWrite, barriers should remain enabled to ensure write durability. Barriers are typically enabled by default, but you can disable barriers using one of the following methods depending on the file system type:

- For `reiserFS`, use the `barrier=none` mount option to disable barriers. To explicitly enable barriers, use `barrier=flush`.
- For `ext3/ext4`, use the `barrier=0` mount option to disable barriers. To explicitly enable barriers, use `barrier=1`.
- For `XFS`, use the `nobarrier` mount option to disable barriers. To explicitly enable barriers, use `barrier`. As of version 4.10 of the mainline Linux kernel, the design of XFS file system always ensures durability. Disabling barriers has no effect and the "nobarrier" option is deprecated. However, some Linux distributions may have backported the changes to a distribution release with an earlier kernel version, check with your distribution vendor for the status in the distribution and version you are running.

## Disk striping

When a high scale VM is attached with several premium storage persistent disks, the disks can be striped together to aggregate their IOPs, bandwidth, and storage capacity.

On Windows, you can use Storage Spaces to stripe disks together. You must configure one column for each disk in a pool. Otherwise, the overall performance of striped volume can be lower than expected, due to uneven distribution of traffic across the disks.

Important: Using Server Manager UI, you can set the total number of columns up to 8 for a striped volume. When attaching more than eight disks, use PowerShell to create the volume. Using PowerShell, you can set the number of columns equal to the number of disks. For example, if there are 16 disks in a single stripe set; specify 16 columns in the `NumberOfColumns` parameter of the `New-VirtualDisk` PowerShell cmdlet.

On Linux, use the MDADM utility to stripe disks together. For detailed steps on striping disks on Linux refer to [Configure Software RAID on Linux](#).

### Stripe Size

An important configuration in disk striping is the stripe size. The stripe size or block size is the smallest chunk of data that application can address on a striped volume. The stripe size you configure depends on the type of application and its request pattern. If you choose the wrong stripe size, it could lead to IO misalignment, which leads to degraded performance of your application.

For example, if an IO request generated by your application is bigger than the disk stripe size, the storage system writes it across stripe unit boundaries on more than one disk. When it is time to access that data, it will have to seek across more than one stripe units to complete the request. The cumulative effect of such behavior

can lead to substantial performance degradation. On the other hand, if the IO request size is smaller than stripe size, and if it is random in nature, the IO requests may add up on the same disk causing a bottleneck and ultimately degrading the IO performance.

Depending on the type of workload your application is running, choose an appropriate stripe size. For random small IO requests, use a smaller stripe size. Whereas for large sequential IO requests use a larger stripe size. Find out the stripe size recommendations for the application you will be running on Premium Storage. For SQL Server, configure stripe size of 64 KB for OLTP workloads and 256 KB for data warehousing workloads. See [Performance best practices for SQL Server on Azure VMs](#) to learn more.

#### NOTE

You can stripe together a maximum of 32 premium storage disks on a DS series VM and 64 premium storage disks on a GS series VM.

## Multi-threading

Azure has designed Premium Storage platform to be massively parallel. Therefore, a multi-threaded application achieves much higher performance than a single-threaded application. A multi-threaded application splits up its tasks across multiple threads and increases efficiency of its execution by utilizing the VM and disk resources to the maximum.

For example, if your application is running on a single core VM using two threads, the CPU can switch between the two threads to achieve efficiency. While one thread is waiting on a disk IO to complete, the CPU can switch to the other thread. In this way, two threads can accomplish more than a single thread would. If the VM has more than one core, it further decreases running time since each core can execute tasks in parallel.

You may not be able to change the way an off-the-shelf application implements single threading or multi-threading. For example, SQL Server is capable of handling multi-CPU and multi-core. However, SQL Server decides under what conditions it will leverage one or more threads to process a query. It can run queries and build indexes using multi-threading. For a query that involves joining large tables and sorting data before returning to the user, SQL Server will likely use multiple threads. However, a user cannot control whether SQL Server executes a query using a single thread or multiple threads.

There are configuration settings that you can alter to influence this multi-threading or parallel processing of an application. For example, in case of SQL Server it is the maximum Degree of Parallelism configuration. This setting called MAXDOP, allows you to configure the maximum number of processors SQL Server can use when parallel processing. You can configure MAXDOP for individual queries or index operations. This is beneficial when you want to balance resources of your system for a performance critical application.

For example, say your application using SQL Server is executing a large query and an index operation at the same time. Let us assume that you wanted the index operation to be more performant compared to the large query. In such a case, you can set MAXDOP value of the index operation to be higher than the MAXDOP value for the query. This way, SQL Server has more number of processors that it can leverage for the index operation compared to the number of processors it can dedicate to the large query. Remember, you do not control the number of threads SQL Server will use for each operation. You can control the maximum number of processors being dedicated for multi-threading.

Learn more about [Degrees of Parallelism](#) in SQL Server. Find out such settings that influence multi-threading in your application and their configurations to optimize performance.

## Queue depth

The queue depth or queue length or queue size is the number of pending IO requests in the system. The value of queue depth determines how many IO operations your application can line up, which the storage disks will be

processing. It affects all the three application performance indicators that we discussed in this article viz., IOPS, throughput, and latency.

Queue Depth and multi-threading are closely related. The Queue Depth value indicates how much multi-threading can be achieved by the application. If the Queue Depth is large, application can execute more operations concurrently, in other words, more multi-threading. If the Queue Depth is small, even though application is multi-threaded, it will not have enough requests lined up for concurrent execution.

Typically, off the shelf applications do not allow you to change the queue depth, because if set incorrectly it will do more harm than good. Applications will set the right value of queue depth to get the optimal performance. However, it is important to understand this concept so that you can troubleshoot performance issues with your application. You can also observe the effects of queue depth by running benchmarking tools on your system.

Some applications provide settings to influence the Queue Depth. For example, the MAXDOP (maximum degree of parallelism) setting in SQL Server explained in previous section. MAXDOP is a way to influence Queue Depth and multi-threading, although it does not directly change the Queue Depth value of SQL Server.

#### *High queue depth*

A high queue depth lines up more operations on the disk. The disk knows the next request in its queue ahead of time. Consequently, the disk can schedule operations ahead of time and process them in an optimal sequence. Since the application is sending more requests to the disk, the disk can process more parallel IOs. Ultimately, the application will be able to achieve higher IOPS. Since application is processing more requests, the total Throughput of the application also increases.

Typically, an application can achieve maximum Throughput with 8-16+ outstanding IOs per attached disk. If a queue depth is one, application is not pushing enough IOs to the system, and it will process less amount of in a given period. In other words, less Throughput.

For example, in SQL Server, setting the MAXDOP value for a query to "4" informs SQL Server that it can use up to four cores to execute the query. SQL Server will determine what is best queue depth value and the number of cores for the query execution.

#### *Optimal queue depth*

Very high queue depth value also has its drawbacks. If queue depth value is too high, the application will try to drive very high IOPS. Unless application has persistent disks with sufficient provisioned IOPS, this can negatively affect application latencies. Following formula shows the relationship between IOPS, latency, and queue depth.

$$\text{IOPS} \times \text{Latency} = \text{Queue Depth}$$

You should not configure Queue Depth to any high value, but to an optimal value, which can deliver enough IOPS for the application without affecting latencies. For example, if the application latency needs to be 1 millisecond, the Queue Depth required to achieve 5,000 IOPS is,  $QD = 5000 \times 0.001 = 5$ .

#### *Queue Depth for Striped Volume*

For a striped volume, maintain a high enough queue depth such that, every disk has a peak queue depth individually. For example, consider an application that pushes a queue depth of 2 and there are four disks in the stripe. The two IO requests will go to two disks and remaining two disks will be idle. Therefore, configure the queue depth such that all the disks can be busy. Formula below shows how to determine the queue depth of striped volumes.

$$\text{QD per Disk} \times \text{No. of Columns per Volume} = \text{QD of Striped Volume}$$

## Throttling

Azure Premium Storage provisions specified number of IOPS and Throughput depending on the VM sizes and disk sizes you choose. Anytime your application tries to drive IOPS or Throughput above these limits of what the VM or disk can handle, Premium Storage will throttle it. This manifests in the form of degraded performance in your application. This can mean higher latency, lower Throughput, or lower IOPS. If Premium Storage does not throttle, your application could completely fail by exceeding what its resources are capable of achieving. So, to avoid performance issues due to throttling, always provision sufficient resources for your application. Take into consideration what we discussed in the VM sizes and Disk sizes sections above. Benchmarking is the best way to figure out what resources you will need to host your application.

## Next steps

If you are looking to benchmark your disk, see our articles on benchmarking a disk:

- For Linux: [Benchmark your application on Azure Disk Storage](#)
- For Windows: [Benchmarking a disk](#).

Learn more about the available disk types:

- For Linux: [Select a disk type](#)
- For Windows: [Select a disk type](#)

For SQL Server users, read articles on Performance Best Practices for SQL Server:

- [Performance Best Practices for SQL Server in Azure Virtual Machines](#)
- [Azure Premium Storage provides highest performance for SQL Server in Azure VM](#)

# Disk performance metrics

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure offers metrics in the Azure portal that provide insight on how your virtual machines (VM) and disks perform. The metrics can also be retrieved through an API call. This article is broken into 3 subsections:

- **Disk IO, throughput and queue depth metrics** - These metrics allow you to see the storage performance from the perspective of a disk and a virtual machine.
- **Disk bursting metrics** - These are the metrics provide observability into our [bursting](#) feature on our premium disks.
- **Storage IO utilization metrics** - These metrics help diagnose bottlenecks in your storage performance with disks.

All metrics are emitted every minute, except for the bursting credit percentage metric, which is emitted every 5 minutes.

## Disk IO, throughput and queue depth metrics

The following metrics are available to get insight on VM and Disk IO, throughput, and queue depth performance:

- **OS Disk Queue Depth**: The number of current outstanding IO requests that are waiting to be read from or written to the OS disk.
- **OS Disk Read Bytes/Sec**: The number of bytes that are read in a second from the OS disk.
- **OS Disk Read Operations/Sec**: The number of input operations that are read in a second from the OS disk.
- **OS Disk Write Bytes/Sec**: The number of bytes that are written in a second from the OS disk.
- **OS Disk Write Operations/Sec**: The number of output operations that are written in a second from the OS disk.
- **Data Disk Queue Depth**: The number of current outstanding IO requests that are waiting to be read from or written to the data disk(s).
- **Data Disk Read Bytes/Sec**: The number of bytes that are read in a second from the data disk(s).
- **Data Disk Read Operations/Sec**: The number of input operations that are read in a second from data disk(s).
- **Data Disk Write Bytes/Sec**: The number of bytes that are written in a second from the data disk(s).
- **Data Disk Write Operations/Sec**: The number of output operations that are written in a second from data disk(s).
- **Disk Read Bytes/Sec**: The number of total bytes that are read in a second from all disks attached to a VM.
- **Disk Read Operations/Sec**: The number of input operations that are read in a second from all disks attached to a VM.
- **Disk Write Bytes/Sec**: The number of bytes that are written in a second from all disks attached to a VM.
- **Disk Write Operations/Sec**: The number of output operations that are written in a second from all disks attached to a VM.

## Bursting metrics

The following metrics help with observability into our [bursting](#) feature on our premium disks:

- **Data Disk Max Burst Bandwidth:** The throughput limit that the data disk(s) can burst up to.
- **OS Disk Max Burst Bandwidth:** The throughput limit that the OS disk can burst up to.
- **Data Disk Max Burst IOPS:** the IOPS limit that the data disk(s) can burst up to.
- **OS Disk Max Burst IOPS:** The IOPS limit that the OS disk can burst up to.
- **Data Disk Target Bandwidth:** The throughput limit that the data(s) disk can achieve without bursting.
- **OS Disk Target Bandwidth:** The throughput limit that the OS disk can achieve without bursting.
- **Data Disk Target IOPS:** The IOPS limit that the data disk(s) can achieve without bursting.
- **OS Disk Target IOPS:** The IOPS limit that the data disk(s) can achieve without bursting.
- **Data Disk Used Burst BPS Credits Percentage:** The accumulated percentage of the throughput burst used for the data disk(s). Emitted on a 5 minute interval.
- **OS Disk Used Burst BPS Credits Percentage:** The accumulated percentage of the throughput burst used for the OS disk. Emitted on a 5 minute interval.
- **Data Disk Used Burst IO Credits Percentage:** The accumulated percentage of the IOPS burst used for the data disk(s). Emitted on a 5 minute interval.
- **OS Disk Used Burst IO Credits Percentage:** The accumulated percentage of the IOPS burst used for the OS disk. Emitted on a 5 minute interval.

## Storage IO utilization metrics

The following metrics help diagnose bottleneck in your Virtual Machine and Disk combination. These metrics are only available with the following configuration:

- Only available on VM series that support premium storage.
- Not available for ultra disks, all other disk types on these VM series can utilize these metrics.

Metrics that help diagnose disk IO capping:

- **Data Disk IOPS Consumed Percentage:** The percentage calculated by the data disk IOPS completed over the provisioned data disk IOPS. If this amount is at 100%, your application running is IO capped from your data disk's IOPS limit.
- **Data Disk Bandwidth Consumed Percentage:** The percentage calculated by the data disk throughput completed over the provisioned data disk throughput. If this amount is at 100%, your application running is IO capped from your data disk's bandwidth limit.
- **OS Disk IOPS Consumed Percentage:** The percentage calculated by the OS disk IOPS completed over the provisioned OS disk IOPS. If this amount is at 100%, your application running is IO capped from your OS disk's IOPS limit.
- **OS Disk Bandwidth Consumed Percentage:** The percentage calculated by the OS disk throughput completed over the provisioned OS disk throughput. If this amount is at 100%, your application running is IO capped from your OS disk's bandwidth limit.

Metrics that help diagnose VM IO capping:

- **VM Cached IOPS Consumed Percentage:** The percentage calculated by the total IOPS completed over the max cached virtual machine IOPS limit. If this amount is at 100%, your application running is IO capped from your VM's cached IOPS limit.
- **VM Cached Bandwidth Consumed Percentage:** The percentage calculated by the total disk throughput completed over the max cached virtual machine throughput. If this amount is at 100%, your application running is IO capped from your VM's cached bandwidth limit.
- **VM uncached IOPS Consumed Percentage:** The percentage calculated by the total IOPS on a virtual machine completed over the max uncached virtual machine IOPS limit. If this amount is at 100%, your application running is IO capped from your VM's uncached IOPS limit.
- **VM Uncached Bandwidth Consumed Percentage:** The percentage calculated by the total disk

throughput on a virtual machine completed over the max provisioned virtual machine throughput. If this amount is at 100%, your application running is IO capped from your VM's uncached bandwidth limit.

## Storage IO metrics example

Let's run through an example of how to use these new Storage IO utilization metrics to help us debug where a bottleneck is in our system. The system setup is the same as the previous example, except this time the attached OS disk is *not* cached.

### Setup:

- Standard\_D8s\_v3
  - Cached IOPS: 16,000
  - Uncached IOPS: 12,800
- P30 OS disk
  - IOPS: 5,000
  - Host caching: **Disabled**
- Two P30 data disks × 2
  - IOPS: 5,000
  - Host caching: **Read/write**
- Two P30 data disks × 2
  - IOPS: 5,000
  - Host caching: **Disabled**

Let's run a benchmarking test on this virtual machine and disk combination that creates IO activity. To learn how to benchmark storage IO on Azure, see [Benchmark your application on Azure Disk Storage](#). From the benchmarking tool, you can see that the VM and disk combination can achieve 22,800 IOPS:

```
demouser@IO-Utilization-Demo:~$ sudo fio fioread.ini
reader1: (g=0): rw=randread, bs=(R) 8192B-8192B, (W) 8192B-8192B, (T) 8192B-8192B, ioengine=libaio, iodepth=256
reader3: (g=0): rw=randread, bs=(R) 8192B-8192B, (W) 8192B-8192B, (T) 8192B-8192B, ioengine=libaio, iodepth=256
reader5: (g=0): rw=randread, bs=(R) 8192B-8192B, (W) 8192B-8192B, (T) 8192B-8192B, ioengine=libaio, iodepth=256
reader10: (g=0): rw=randread, bs=(R) 8192B-8192B, (W) 8192B-8192B, (T) 8192B-8192B, ioengine=libaio, iodepth=256
reader12: (g=0): rw=randread, bs=(R) 8192B-8192B, (W) 8192B-8192B, (T) 8192B-8192B, ioengine=libaio, iodepth=256
fio-3.1
Starting 5 processes
Jobs: 5 (f=5), 0-22800 IOPS: [r(5)][0.4%][r=178MiB/s,w=0KiB/s] r=22.8k, v=0 IOPS][eta 27m:06s]
```

The Standard\_D8s\_v3 can achieve a total of 28,600 IOPS. Using the metrics, let's investigate what's going on and identify our storage IO bottleneck. On the left pane, select **Metrics**:

The screenshot shows the Azure portal interface for a virtual machine named "IO-Utilization-Demo". The left sidebar has a "Metrics" icon highlighted with a red box. The main content area shows the "Metrics" blade with the following details:

- Monitoring**: Shows an advisor message: "Adviser (1 of 6): All network ports should be restricted on network security groups associated to [redacted]."
- Essentials**:
  - Resource group: (change)
  - Status: Running
  - Location: East US 2
  - Subscription: (change)
  - Subscription ID: [redacted]
  - Tags: (change) Click here to add tags
- Metrics**: A list of metrics including:
  - Cached IOPS Consumed Percentage
  - Uncached IOPS Consumed Percentage
  - Latency
  - Throughput
  - Throughput Latency
  - Throughput Throughput
- Diagnostic settings**
- Logs**
- Connection monitor**
- Automation**

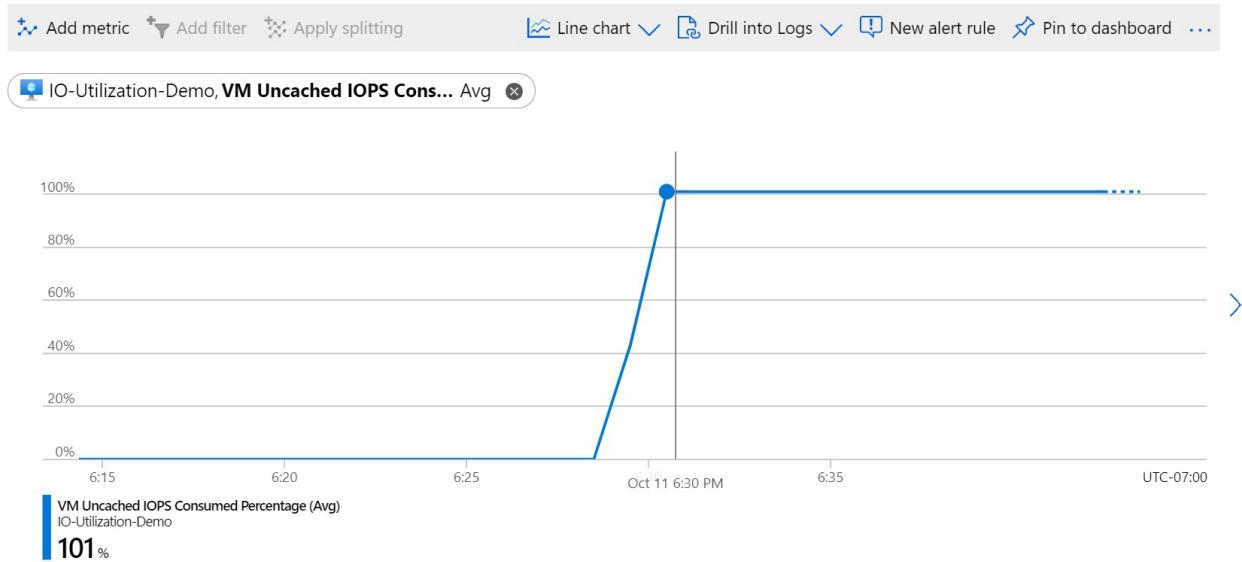
Let's first take a look at our **VM Cached IOPS Consumed Percentage** metric:

### Avg VM Cached IOPS Consumed Percentage for IO-Utilization-Demo



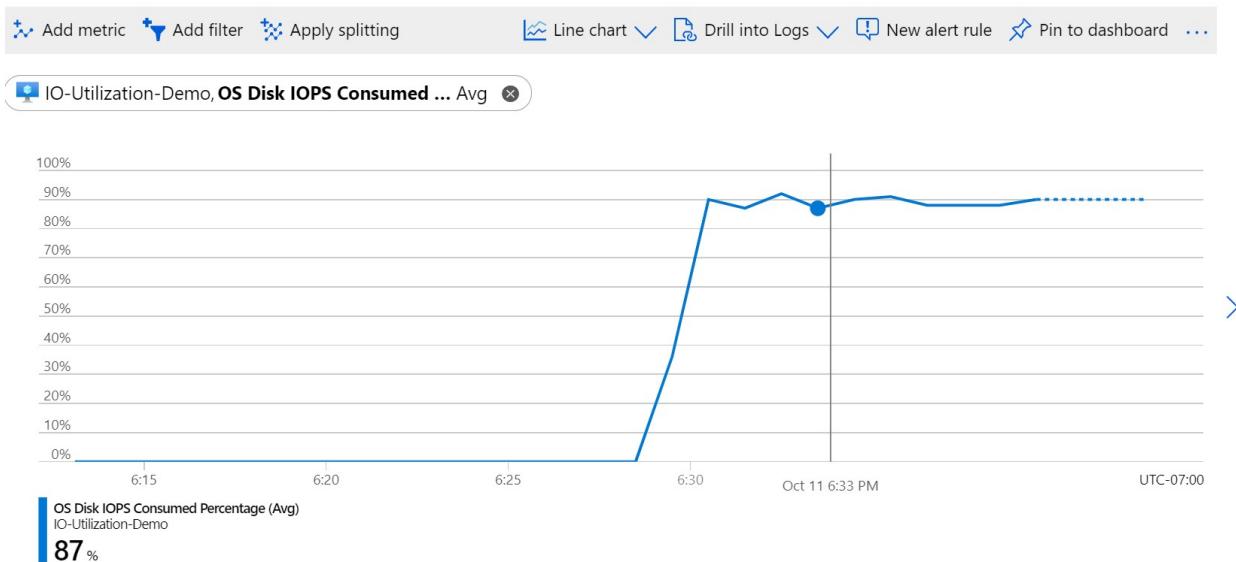
This metric tells us that 61% of the 16,000 IOPS allotted to the cached IOPS on the VM is being used. This percentage means that the storage IO bottleneck isn't with the disks that are cached because it isn't at 100%. Now let's look at our **VM Uncached IOPS Consumed Percentage** metric:

### Avg VM Uncached IOPS Consumed Percentage for IO-Utilization-Demo



This metric is at 100%. It tells us that all of the 12,800 IOPS allotted to the uncached IOPS on the VM are being used. One way we can remediate this issue is to change the size of our VM to a larger size that can handle the additional IO. But before we do that, let's look at the attached disk to find out how many IOPS they are seeing. Check the OS Disk by looking at the **OS Disk IOPS Consumed Percentage**:

### Avg OS Disk IOPS Consumed Percentage for IO-Utilization-Demo



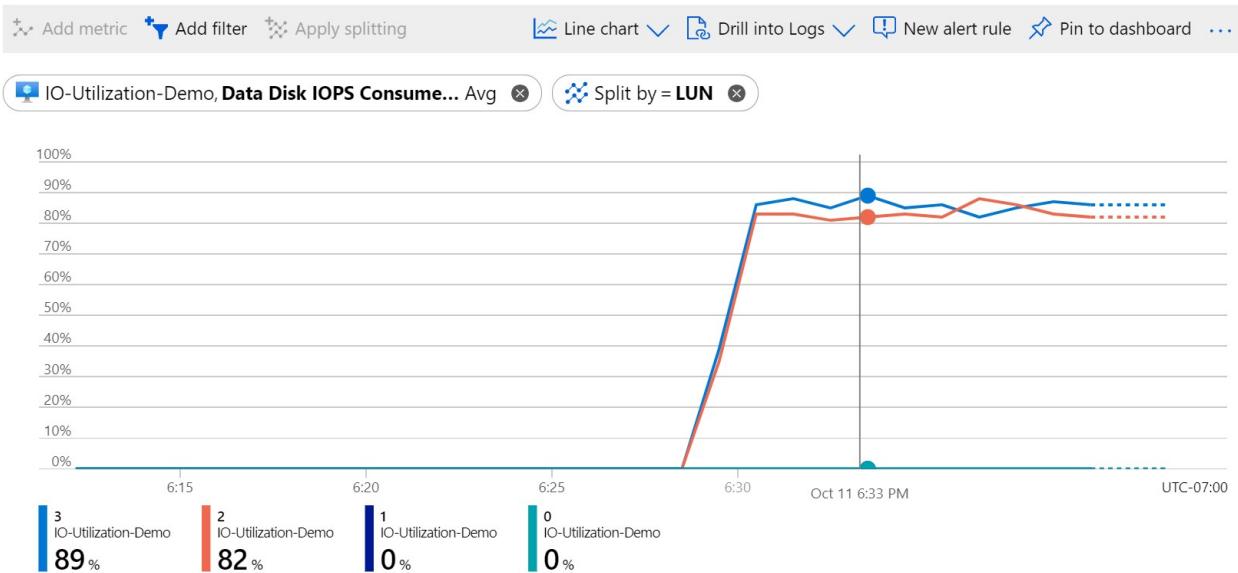
This metric tells us that around 90% of the 5,000 IOPS provisioned for this P30 OS disk is being used. This percentage means there's no bottleneck at the OS disk. Now let's check the data disks that are attached to the VM by looking at the **Data Disk IOPS Consumed Percentage**:

### Avg Data Disk IOPS Consumed Percentage for IO-Utilization-Demo

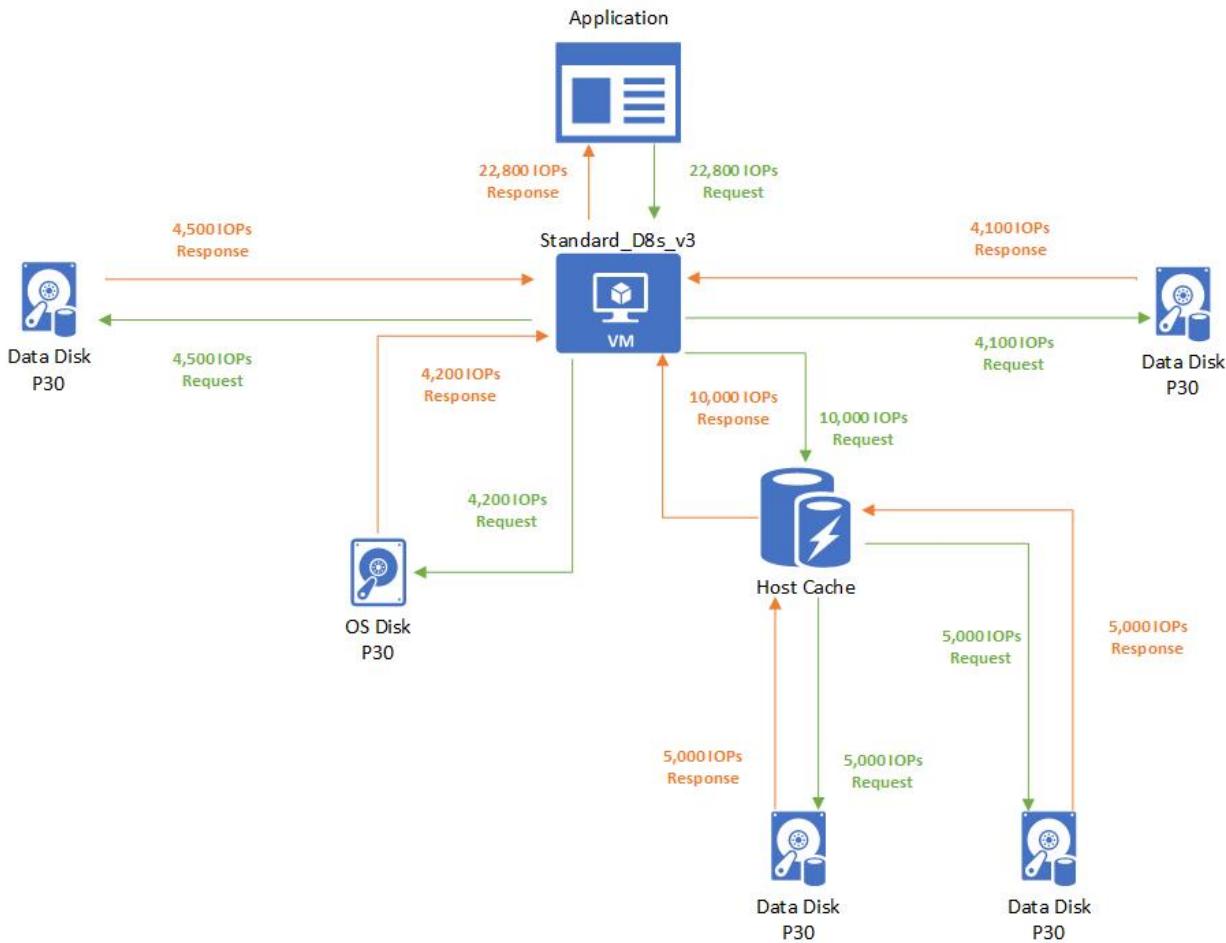


This metric tells us that the average IOPS consumed percentage across all the disks attached is around 42%. This percentage is calculated based on the IOPS that are used by the disks, and aren't being served from the host cache. Let's drill deeper into this metric by applying *splitting* on these metrics and splitting by the LUN value:

## Avg Data Disk IOPS Consumed Percentage for IO-Utilization-Demo by LUN



This metric tells us the data disks attached on LUN 3 and 2 are using around 85% of their provisioned IOPS. Here is a diagram of what the IO looks like from the VM and disks architecture:



## Next steps

- [Azure Monitor Metrics overview](#)
- [Metrics aggregation explained](#)
- [Create, view, and manage metric alerts using Azure Monitor](#)

# Managed disk bursting

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure offers the ability to boost disk storage IOPS and MB/s performance, this is referred to as bursting for both virtual machines (VM) and disks. You can effectively use VM and disk bursting to achieve better bursting performance on both your VMs and disk.

Bursting for Azure VMs and disk resources aren't dependent on each other. You don't need to have a burst-capable VM for an attached burst-capable disk to burst. Similarly, you don't need to have a burst-capable disk attached to your burst-capable VM for the VM to burst.

## Common scenarios

The following scenarios can benefit greatly from bursting:

- **Improve startup times** – With bursting, your instance will startup at a faster rate. For example, the default OS disk for premium enabled VMs is the P4 disk, which is a provisioned performance of up to 120 IOPS and 25 MB/s. With bursting, the P4 can go up to 3500 IOPS and 170 MB/s allowing for startup to accelerate by up to 6X.
- **Handle batch jobs** – Some application workloads are cyclical in nature. They require a baseline performance most of the time, and higher performance for short periods of time. An example of this is an accounting program that processes daily transactions that require a small amount of disk traffic. At the end of the month this program would complete reconciling reports that need a much higher amount of disk traffic.
- **Traffic spikes** – Web servers and their applications can experience traffic surges at any time. If your web server is backed by VMs or disks that use bursting, the servers would be better equipped to handle traffic spikes.

## Disk-level bursting

Currently, there are two managed disk types that can burst, [premium SSDs](#), and [standard SSDs](#). Other disk types cannot currently burst. There are two models of bursting for disks:

- An on-demand bursting model, where the disk bursts whenever its needs exceed its current capacity. This model incurs additional charges anytime the disk bursts. On-demand bursting is only available for premium SSDs larger than 512 GiB.
- A credit-based model, where the disk will burst only if it has burst credits accumulated in its credit bucket. This model does not incur additional charges when the disk bursts. Credit-based bursting is only available for premium SSDs 512 GiB and smaller, and standard SSDs 1024 GiB and smaller.

Azure [premium SSDs](#) can use either bursting model, but [standard SSDs](#) currently only offer credit-based bursting.

Additionally, the [performance tier of managed disks can be changed](#), which could be ideal if your workload would otherwise be running in burst.

	CREDIT-BASED BURSTING	ON-DEMAND BURSTING	CHANGING PERFORMANCE TIER
Scenarios	Ideal for short-term scaling (30 minutes or less).	Ideal for short-term scaling(Not time restricted).	Ideal if your workload would otherwise continually be running in burst.
Cost	Free	Cost is variable, see the <a href="#">Billing</a> section for details.	The cost of each performance tier is fixed, see <a href="#">Managed Disks pricing</a> for details.
Availability	Only available for premium SSDs 512 GiB and smaller, and standard SSDs 1024 GiB and smaller.	Only available for premium SSDs larger than 512 GiB.	Available to all premium SSD sizes.
Enablement	Enabled by default on eligible disks.	Must be enabled by user.	User must manually change their tier.

## On-demand bursting

Premium SSDs using the on-demand bursting model of disk bursting can burst beyond original provisioned targets, as often as needed by their workload, up to the max burst target. For example, on a 1-TiB P30 disk, the provisioned IOPS is 5000 IOPS. When disk bursting is enabled on this disk, your workloads can issue IOs to this disk up to the max burst performance of 30,000 IOPS and 1,000 MBps. For the max burst targets on each supported disk, see [Scalability and performance targets for VM disks](#).

If you expect your workloads to frequently run beyond the provisioned perf target, disk bursting won't be cost-effective. In this case, we recommend that you change your disk's performance tier to a [higher tier](#) instead, for better baseline performance. Review your billing details and assess that against the traffic pattern of your workloads.

Before you enable on-demand bursting, understand the following:

- On-demand bursting cannot be enabled on a premium SSD that has less than or equal to 512 GiB. Premium SSDs less than or equal to 512 GiB will always use credit-based bursting.
- On-demand bursting is only supported on premium SSDs. If a premium SSD with on-demand bursting enabled is switched to another disk type, then disk bursting is disabled.
- On-demand bursting doesn't automatically disable itself when the performance tier is changed. If you want to change your performance tier but do not want to keep disk bursting, you must disable it.
- On-demand bursting can only be enabled when the disk is detached from a VM or when the VM is stopped. On-demand bursting can be disabled 12 hours after it has been enabled.

## Regional availability

Currently, the on-demand model for disk bursting is available in all public Azure regions.

## Billing

Premium SSDs using the on-demand bursting model are charged an hourly burst enablement flat fee and transaction costs apply to any burst transactions beyond the provisioned target. Transaction costs are charged using the pay-as-you go model, based on uncached disk IOs, including both reads and writes that exceed provisioned targets. The following is an example of disk traffic patterns over a billing hour:

Disk configuration: Premium SSD – 1 TiB (P30), Disk bursting enabled.

- 00:00:00 – 00:10:00 Disk IOPS below provisioned target of 5,000 IOPS
- 00:10:01 – 00:10:10 Application issued a batch job causing the disk IOPS to burst at 6,000 IOPS for 10 seconds.

seconds

- 00:10:11 – 00:59:00 Disk IOPS below provisioned target of 5,000 IOPS
- 00:59:01 – 01:00:00 Application issued another batch job causing the disk IOPS to burst at 7,000 IOPS for 60 seconds

In this billing hour, the cost of bursting consists of two charges:

The first charge is the burst enablement flat fee of \$X (determined by your region). This flat fee is always charged on the disk regardless of the attach status until it is disabled.

Second is the burst transaction cost. Disk bursting occurred in two time slots. From 00:10:01 – 00:10:10, the accumulated burst transaction is  $(6,000 - 5,000) \times 10 = 10,000$ . From 00:59:01 – 01:00:00, the accumulated burst transaction is  $(7,000 - 5,000) \times 60 = 120,000$ . The total burst transactions are  $10,000 + 120,000 = 130,000$ . Burst transaction cost will be charged at \$Y based on 13 units of 10,000 transactions (based on regional pricing).

With that, the total cost on disk bursting of this billing hour equals to  $\$X + \$Y$ . The same calculation would apply for bursting over provisioned target of MBps. We translate the usage of MB to transactions with IO size of 256KB. If your disk traffic exceed both provisioned IOPS and MBps target, you can refer to the example below to calculate the burst transactions.

Disk configuration: Premium SSD – 1 TB (P30), Disk bursting enabled.

- 00:00:01 – 00:00:05 Application issued a batch job causing the disk IOPS to burst at 10,000 IOPS and 300 MBps for five seconds.
- 00:00:06 – 00:00:10 Application issued a recovery job causing the disk IOPS to burst at 6,000 IOPS and 600 MBps for five seconds.

The burst transaction is accounted as the max number of transactions from either IOPS or MBps bursting. From 00:00:01 – 00:00:05, the accumulated burst transaction is  $\max((10,000 - 5,000), (300 - 200) * 1024 / 256) * 5 = 25,000$  transactions. From 00:00:06 – 00:00:10, the accumulated burst transaction is  $\max((6,000 - 5,000), (600 - 200) * 1024 / 256) * 5 = 8,000$  transactions. On top of that, you include the burst enablement flat fee to get the total cost for enabling on-demand based disk bursting.

You can refer to the [Managed Disks pricing page](#) for details on pricing and use [Azure Pricing Calculator](#) to make the assessment for your workload.

To enable on-demand bursting, see [Enable on-demand bursting](#).

### Credit-based bursting

For premium SSDs, credit-based bursting is available for disk sizes P20 and smaller. For standard SSDs, credit-based bursting is available for disk sizes E30 and smaller. For both standard and premium SSDs, credit-based bursting is available in all regions in Azure Public, Government, and China Clouds. By default, disk bursting is enabled on all new and existing deployments of supported disk sizes. VM-level bursting only uses credit-based bursting.

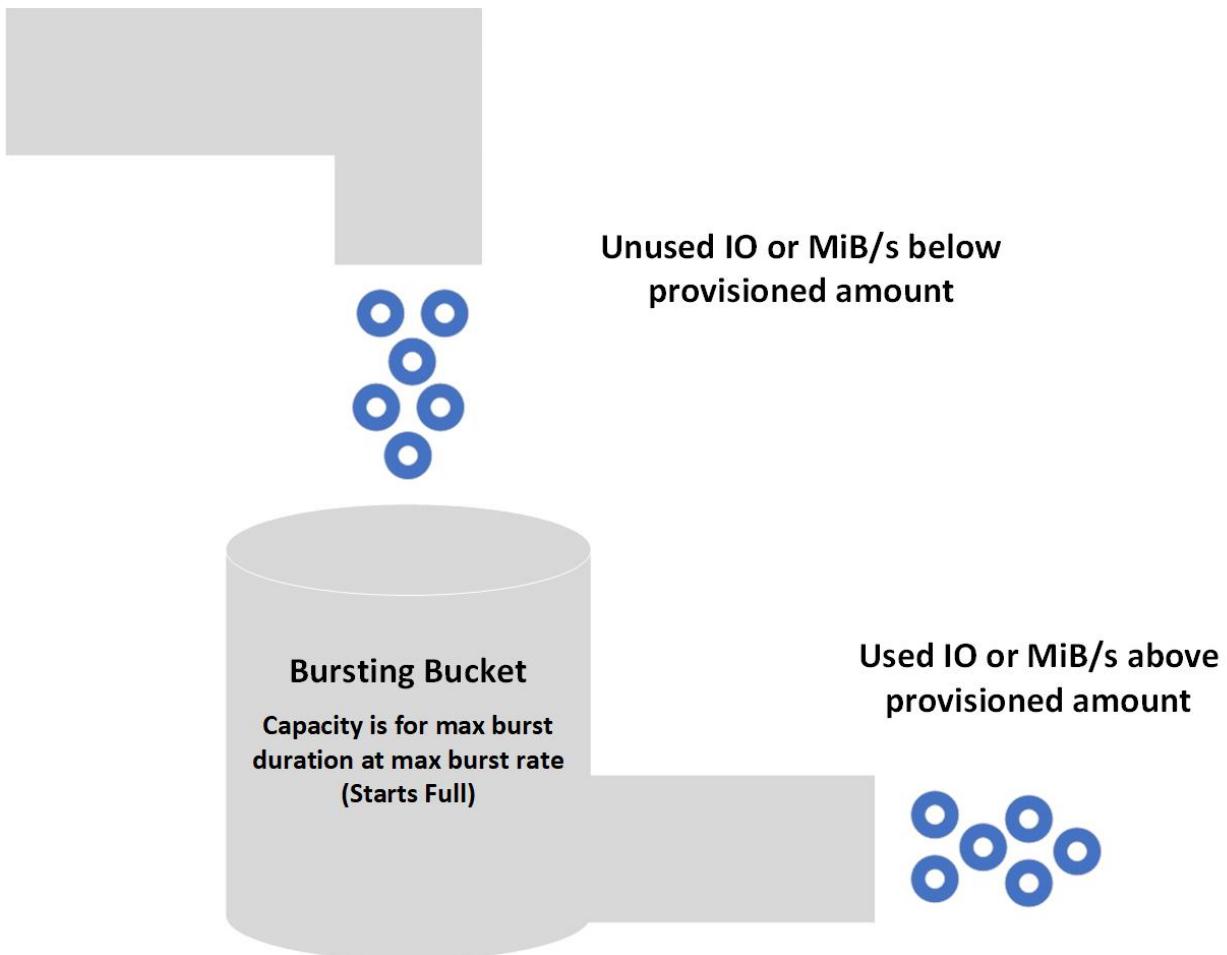
## Virtual machine-level bursting

VM-level bursting only uses the credit-based model for bursting, it is enabled by default for most Premium Storage supported VMs.

## Bursting flow

The bursting credit system applies in the same manner at both the VM level and disk level. Your resource, either a VM or disk, will start with fully stocked credits in its own burst bucket. These credits allow you to burst for up to 30 minutes at the maximum burst rate. You accumulate credits whenever the resource's IOPS or MB/s are

being utilized below the resource's performance target. If your resource has accrued bursting credits and your workload needs the extra performance, your resource can use those credits to go above its performance limits and increase its performance to meet the workload demands.



How you spend your available credits is up to you. You can use your 30 minutes of burst credits consecutively or sporadically throughout the day. When resources are deployed they come with a full allocation of credits. When those deplete, it takes less than a day to restock. Credits can be spent at your discretion, the burst bucket does not need to be full in order for resources to burst. Burst accumulation varies depending on each resource, since it is based on unused IOPS and MB/s below their performance targets. Higher baseline performance resources can accrue their bursting credits faster than lower baseline performing resources. For example, a P1 disk idling will accrue 120 IOPS per second, whereas an idling P20 disk would accrue 2,300 IOPS per second.

## Bursting states

There are three states your resource can be in with bursting enabled:

- **Accruing** – The resource's IO traffic is using less than the performance target. Accumulating bursting credits for IOPS and MB/s are done separate from one another. Your resource can be accruing IOPS credits and spending MB/s credits or vice versa.
- **Bursting** – The resource's traffic is using more than the performance target. The burst traffic will independently consume credits from IOPS or bandwidth.
- **Constant** – The resource's traffic is exactly at the performance target.

## Bursting examples

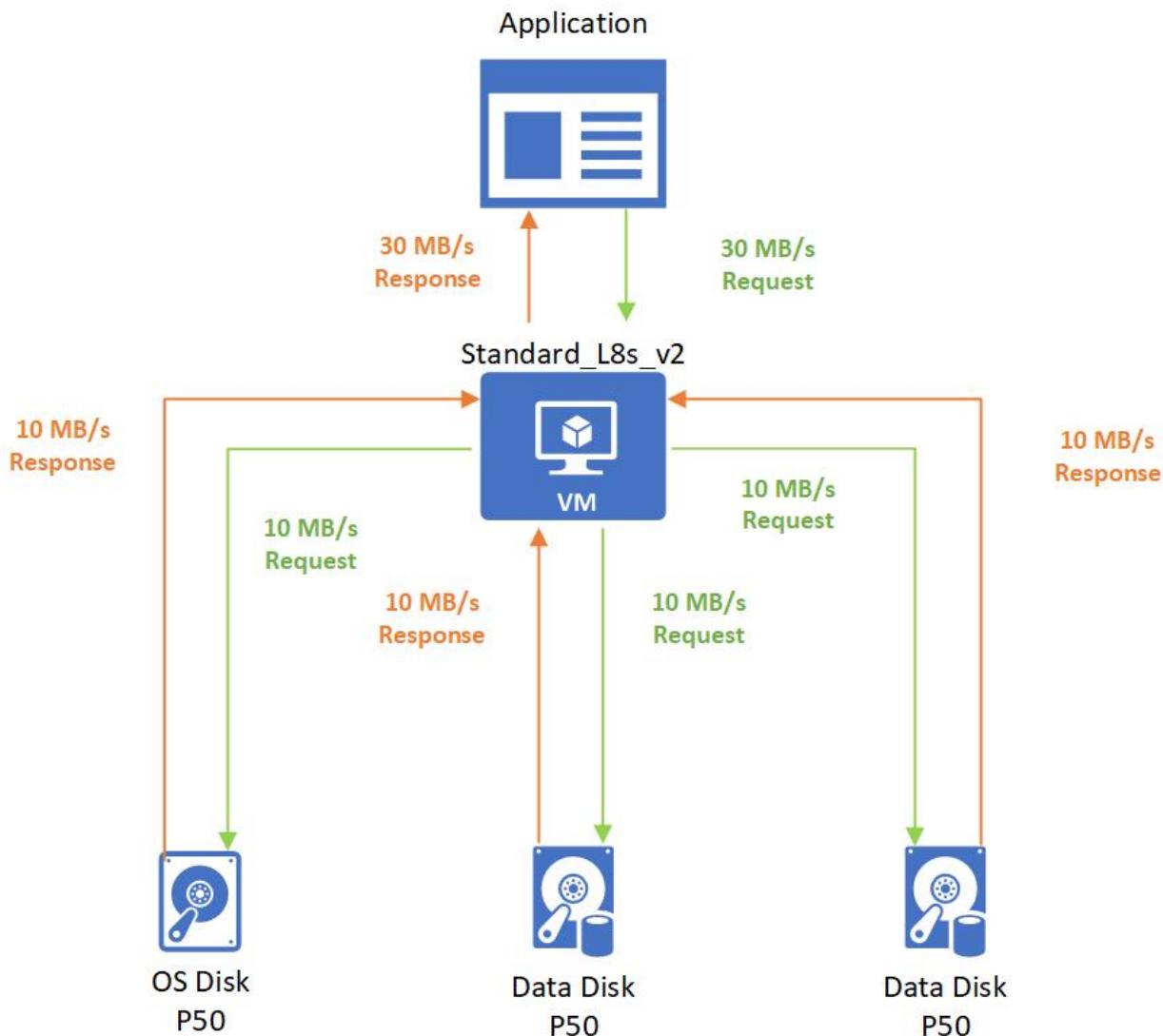
The following examples show how bursting works with various VM and disk combinations. To make the examples easy to follow, we will focus on MB/s, but the same logic is applied independently to IOPS.

### Burstable virtual machine with non-burstable disks

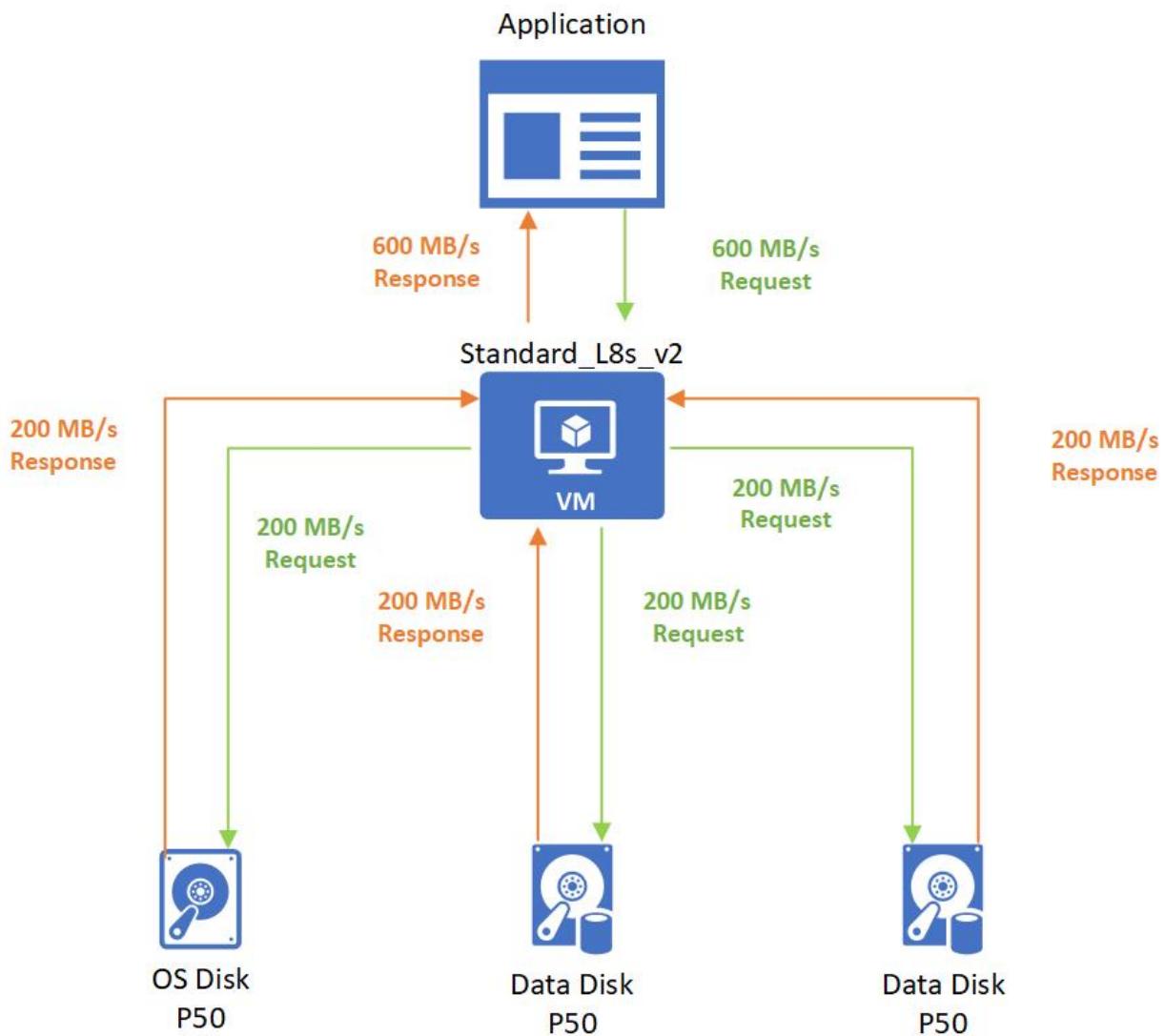
## VM and disk combination:

- Standard\_L8s\_v2
  - Uncached MB/s: 160
  - Max burst MB/s: 1,280
- P50 OS Disk
  - Provisioned MB/s: 250
  - On-Demand Bursting: **not enabled**
- 2 P50 Data Disks
  - Provisioned MB/s: 250
  - On-Demand Bursting: **not enabled**

After the initial boot up, an application is run on the VM and has a non-critical workload. This workload requires 30 MB/s that gets spread evenly across all the disks.



Then the application needs to process a batched job that requires 600 MB/s. The Standard\_L8s\_v2 bursts to meet this demand and then requests to the disks get evenly spread out to P50 disks.

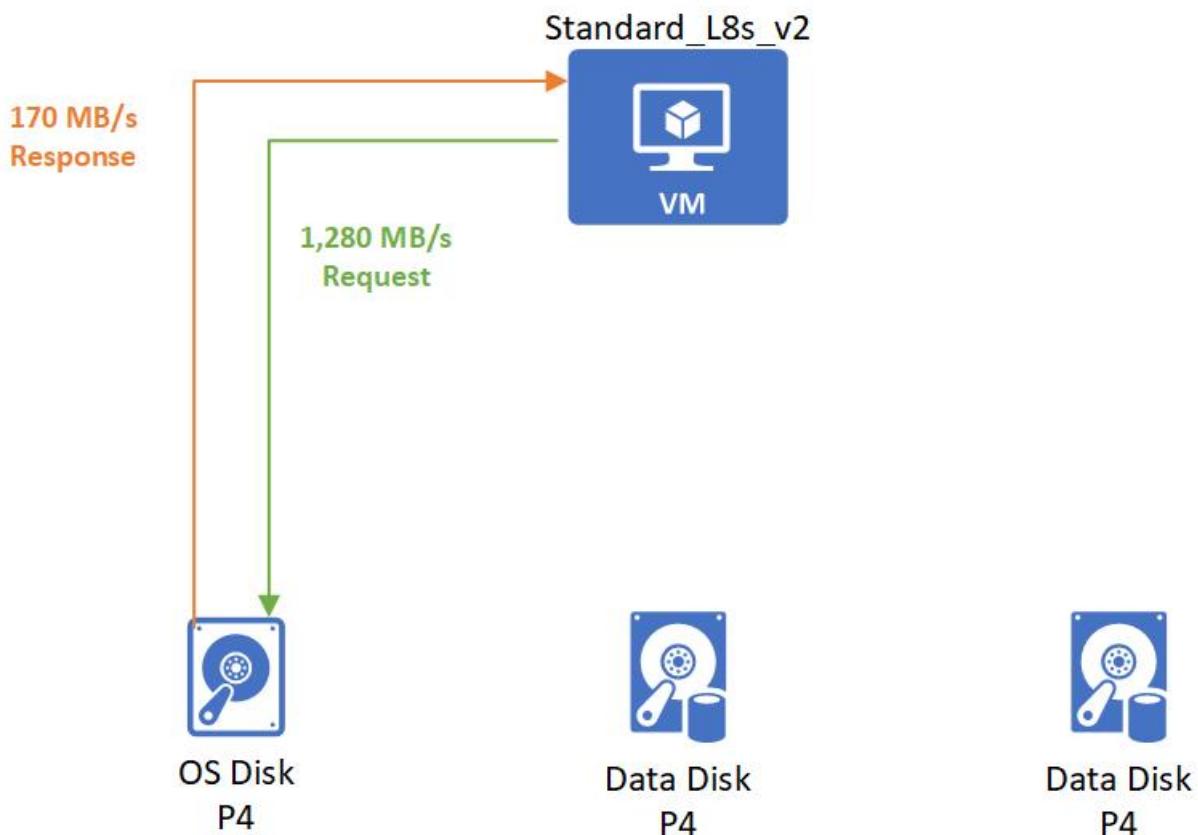


### Burstable virtual machine with burstable disks

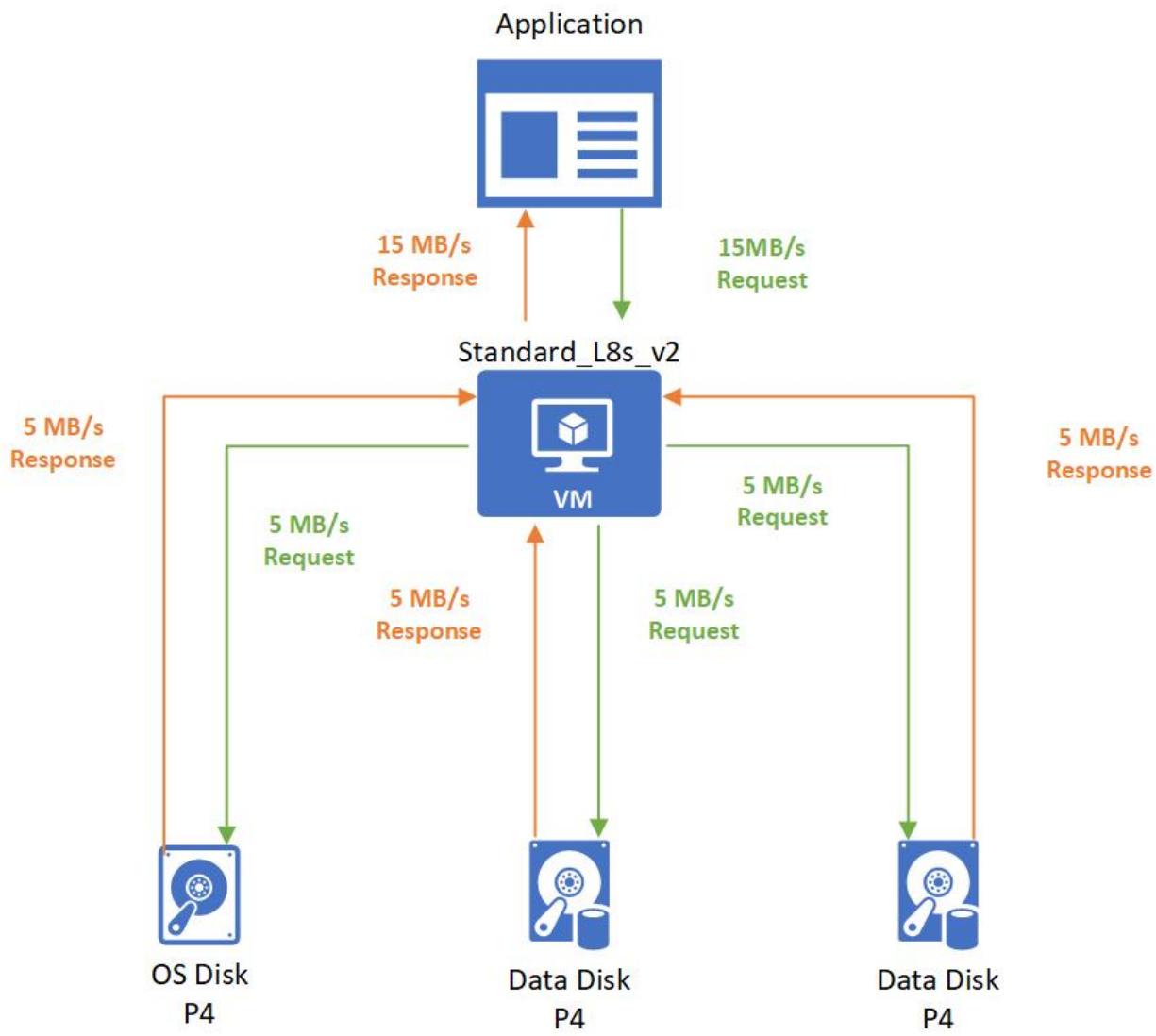
VM and disk combination:

- Standard\_L8s\_v2
  - Uncached MB/s: 160
  - Max burst MB/s: 1,280
- P4 OS Disk
  - Provisioned MB/s: 25
  - Max burst MB/s: 170
- 2 P4 Data Disks
  - Provisioned MB/s: 25
  - Max burst MB/s: 170

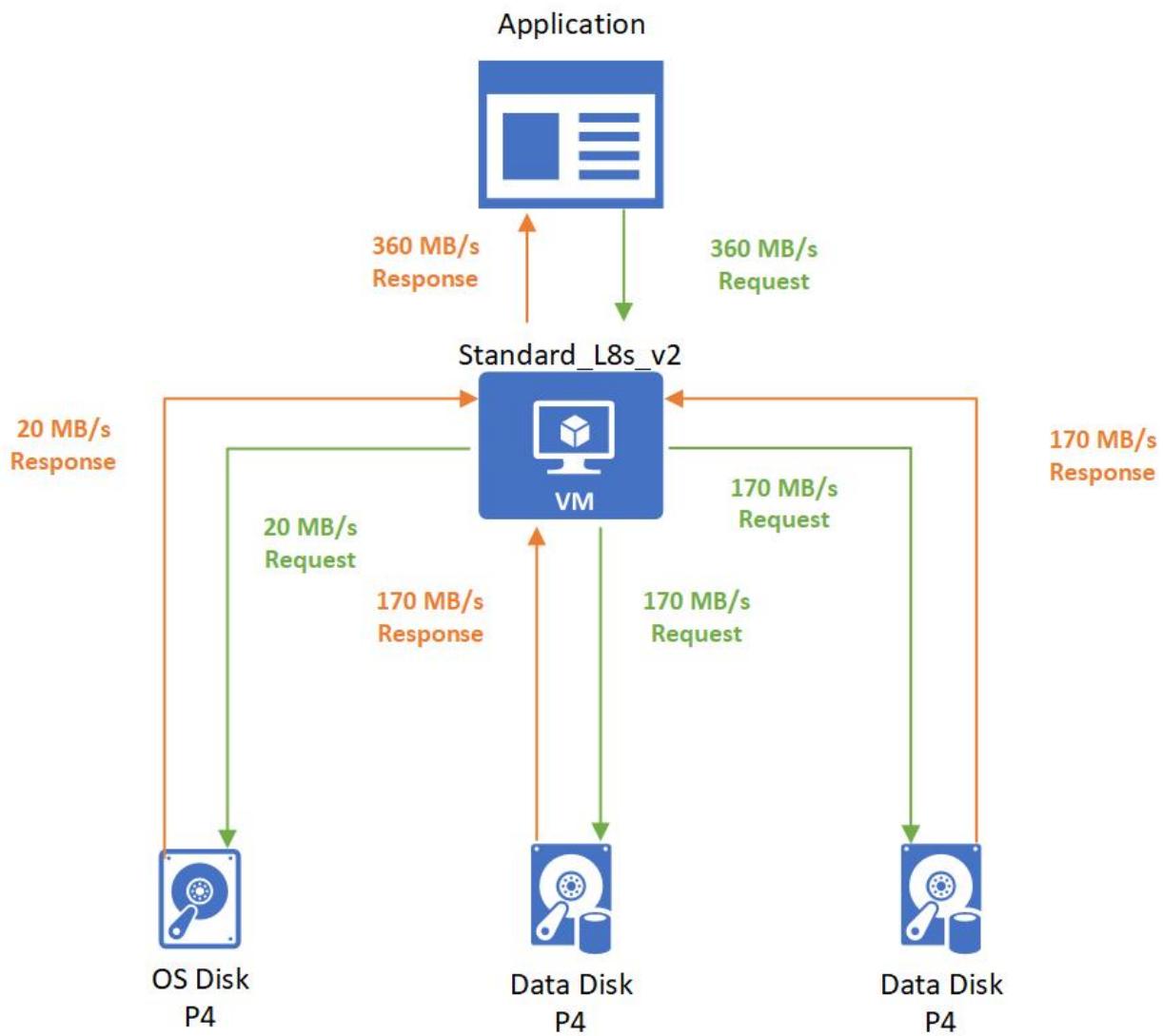
When the VM starts, it will burst to request its burst limit of 1,280 MB/s from the OS disk and the OS disk will respond with its burst performance of 170 MB/s.



After startup, you start an application that has a non-critical workload. This application requires 15 MB/s that gets spread evenly across all the disks.



Then the application needs to process a batched job that requires 360 MB/s. The Standard\_L8s\_v2 bursts to meet this demand and then requests. Only 20 MB/s are needed by the OS disk. The remaining 340 MB/s are handled by the bursting P4 data disks.



## Next steps

- To enable on-demand bursting, see [Enable on-demand bursting](#).
- To learn how to gain insight into your bursting resources, see [Disk bursting metrics](#).
- To see exactly how much each applicable disk size can burst, see [Scalability and performance targets for VM disks](#).

# Enable on-demand bursting

9/21/2022 • 3 minutes to read • [Edit Online](#)

Premium solid-state drives (SSD) have two available bursting models; credit-based bursting and on-demand bursting. This article covers how to switch to on-demand bursting. Disks that use the on-demand model can burst beyond their original provisioned targets. On-demand bursting occurs as often as needed by the workload, up to the maximum burst target. On-demand bursting incurs additional charges.

For details on disk bursting, see [Managed disk bursting](#).

For the max burst targets on each supported disk, see [Scalability and performance targets for VM disks](#).

## IMPORTANT

You don't need to follow the steps in this article to use credit-based bursting. By default, credit-based bursting is enabled on all eligible disks.

Before you enable on-demand bursting, understand the following:

- On-demand bursting cannot be enabled on a premium SSD that has less than or equal to 512 GiB. Premium SSDs less than or equal to 512 GiB will always use credit-based bursting.
- On-demand bursting is only supported on premium SSDs. If a premium SSD with on-demand bursting enabled is switched to another disk type, then disk bursting is disabled.
- On-demand bursting doesn't automatically disable itself when the performance tier is changed. If you want to change your performance tier but do not want to keep disk bursting, you must disable it.
- On-demand bursting can only be enabled when the disk is detached from a VM or when the VM is stopped. On-demand bursting can be disabled 12 hours after it has been enabled.

## Regional availability

Currently, the on-demand model for disk bursting is available in all public Azure regions.

## Get started

On-demand bursting can be enabled with either the Azure PowerShell module, the Azure CLI, or Azure Resource Manager templates. The following examples cover how to create a new disk with on-demand bursting enabled and enabling on-demand bursting on existing disks.

- [PowerShell](#)
- [Azure CLI](#)
- [Azure Resource Manager](#)

On-demand bursting cmdlets are available in version 5.5.0 and newer of the Az module. Alternatively, you may use the [Azure Cloud Shell](#).

### Create an empty data disk with on-demand bursting

A managed disk must be larger than 512 GiB to enable on-demand bursting. Replace the `<myResourceGroupDisk>` and `<myDataDisk>` parameters then run the following script to create a premium SSD with on-demand bursting:

```
Set-AzContext -SubscriptionName <yourSubscriptionName>

$diskConfig = New-AzDiskConfig -Location 'WestCentralUS' -CreateOption Empty -DiskSizeGB 1024 -SkuName Premium_LRS -BurstingEnabled $true

$dataDisk = New-AzDisk -ResourceGroupName <myResourceGroupDisk> -DiskName <myDataDisk> -Disk $diskConfig
```

## Enable on-demand bursting on an existing disk

A managed disk must be larger than 512 GiB to enable on-demand bursting. Replace the `<myResourceGroupDisk>`, `<myDataDisk>` parameters and run this command to enable on-demand bursting on an existing disk:

```
New-AzDiskUpdateConfig -BurstingEnabled $true | Update-AzDisk -ResourceGroupName <myResourceGroupDisk> -DiskName <myDataDisk> //Set the flag to $false to disable on-demand bursting
```

## Next steps

To learn how to gain insight into your bursting resources, see [Disk bursting metrics](#).

# Performance tiers for managed disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The performance of your Azure managed disk is set when you create your disk, in the form of its performance tier. When you set the provisioned size of your disk, a performance tier is automatically selected. The performance tier determines the IOPS and throughput your managed disk has. The performance tier can be changed at deployment or afterwards, without changing the size of the disk and without downtime.

Changing the performance tier allows you to prepare for and meet higher demand without using your disk's bursting capability. It can be more cost-effective to change your performance tier rather than rely on bursting, depending on how long the additional performance is necessary. This is ideal for events that temporarily require a consistently higher level of performance, like holiday shopping, performance testing, or running a training environment. To handle these events, you can switch a disk to a higher performance tier without downtime, for as long as you need the additional performance. You can then return to the original tier without downtime when the additional performance is no longer necessary.

## Restrictions

- This feature is currently supported only for premium SSDs.
- This feature isn't currently supported with shared disks.
- The P60, P70, and P80 performance tiers can only be used by disks that are larger than 4,096 GiB.
- A disk's performance tier can be downgraded only once every 12 hours.
- The system does not return Performance Tier for disks created before June 2020. You can take advantage of Performance Tier for an older disk by updating it with the baseline Tier.

## How it works

When you first deploy or provision a disk, the baseline performance tier for that disk is set based on the provisioned disk size. You can use a performance tier higher than the original baseline to meet higher demand. When you no longer need that performance level, you can return to the initial baseline performance tier.

### Billing impact

Your billing changes as your performance tier changes. For example, if you provision a P10 disk (128 GiB), your baseline performance tier is set as P10 (500 IOPS and 100 MBps). You'll be billed at the P10 rate. You can upgrade the tier to match the performance of P50 (7,500 IOPS and 250 MBps) without increasing the disk size. During the time of the upgrade, you'll be billed at the P50 rate. When you no longer need the higher performance, you can return to the P10 tier. The disk will once again be billed at the P10 rate.

For billing information, see [Managed disk pricing](#).

## What tiers can be changed

The following table depicts which tiers each baseline performance tier can upgrade to.

DISK SIZE	BASELINE PERFORMANCE TIER	CAN BE UPGRADED TO
4 GiB	P1	P2, P3, P4, P6, P10, P15, P20, P30, P40, P50

DISK SIZE	BASELINE PERFORMANCE TIER	CAN BE UPGRADED TO
8 GiB	P2	P3, P4, P6, P10, P15, P20, P30, P40, P50
16 GiB	P3	P4, P6, P10, P15, P20, P30, P40, P50
32 GiB	P4	P6, P10, P15, P20, P30, P40, P50
64 GiB	P6	P10, P15, P20, P30, P40, P50
128 GiB	P10	P15, P20, P30, P40, P50
256 GiB	P15	P20, P30, P40, P50
512 GiB	P20	P30, P40, P50
1 TiB	P30	P40, P50
2 TiB	P40	P50
4 TiB	P50	None
8 TiB	P60	P70, P80
16 TiB	P70	P80
32 TiB	P80	None

## Next steps

To learn how to change your performance tier, see [portal](#) or [PowerShell/CLI](#) articles.

# Change your performance tier without downtime using the Azure PowerShell module or the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

The performance of your Azure managed disk is set when you create your disk, in the form of its performance tier. The performance tier determines the IOPS and throughput your managed disk has. When you set the provisioned size of your disk, a performance tier is automatically selected. The performance tier can be changed at deployment or afterwards, without changing the size of the disk and without downtime. To learn more about performance tiers, see [Performance tiers for managed disks](#).

Changing your performance tier has billing implications. See [Billing impact](#) for details.

## Restrictions

- This feature is currently supported only for premium SSDs.
- This feature isn't currently supported with shared disks.
- The P60, P70, and P80 performance tiers can only be used by disks that are larger than 4,096 GiB.
- A disk's performance tier can be downgraded only once every 12 hours.
- The system does not return Performance Tier for disks created before June 2020. You can take advantage of Performance Tier for an older disk by updating it with the baseline Tier.

## Prerequisites

- [Azure CLI](#)
- [PowerShell](#)

Install the latest [Azure CLI](#) and sign in to an Azure account with [az login](#).

## Create an empty data disk with a tier higher than the baseline tier

- [Azure CLI](#)
- [PowerShell](#)

```
subscriptionId=<yourSubscriptionIDHere>
resourceGroupName=<yourResourceGroupNameHere>
diskName=<yourDiskNameHere>
diskSize=<yourDiskSizeHere>
performanceTier=<yourDesiredPerformanceTier>
region=westcentralus

az account set --subscription $subscriptionId

az disk create -n $diskName -g $resourceGroupName -l $region --sku Premium_LRS --size-gb $diskSize --tier
$performanceTier
```

## Create an OS disk with a tier higher than the baseline tier from an Azure Marketplace image

```
resourceGroupName=<yourResourceGroupNameHere>
diskName=<yourDiskNameHere>
performanceTier=<yourDesiredPerformanceTier>
region=westcentralus
image=Canonical:UbuntuServer:18.04-LTS:18.04.202002180

az disk create -n $diskName -g $resourceGroupName -l $region --image-reference $image --sku Premium_LRS --
tier $performanceTier
```

## Update the tier of a disk without downtime

- [Azure CLI](#)
- [PowerShell](#)

1. Update the tier of a disk even when it is attached to a running VM

```
resourceGroupName=<yourResourceGroupNameHere>
diskName=<yourDiskNameHere>
performanceTier=<yourDesiredPerformanceTier>

az disk update -n $diskName -g $resourceGroupName --set tier=$performanceTier
```

## Show the tier of a disk

- [Azure CLI](#)
- [PowerShell](#)

```
az disk show -n $diskName -g $resourceGroupName --query [tier] -o tsv
```

## Next steps

If you need to resize a disk to take advantage of the higher performance tiers, see these articles:

- [Expand virtual hard disks on a Linux VM with the Azure CLI](#)
- [Expand a managed disk attached to a Windows virtual machine](#)

# Change your performance tier using the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

The performance of your Azure managed disk is set when you create your disk, in the form of its performance tier. The performance tier determines the IOPS and throughput your managed disk has. When you set the provisioned size of your disk, a performance tier is automatically selected. The performance tier can be changed at deployment or afterwards, without changing the size of the disk and without downtime. To learn more about performance tiers, see [Performance tiers for managed disks](#).

Changing your performance tier has billing implications. See [Billing impact](#) for details.

## Restrictions

- This feature is currently supported only for premium SSDs.
- This feature isn't currently supported with shared disks.
- The P60, P70, and P80 performance tiers can only be used by disks that are larger than 4,096 GiB.
- A disk's performance tier can be downgraded only once every 12 hours.
- The system does not return Performance Tier for disks created before June 2020. You can take advantage of Performance Tier for an older disk by updating it with the baseline Tier.

## Getting started

### New disks

The following steps show how to change the performance tier of your disk when you first create the disk:

1. Sign in to the [Azure portal](#).
2. Navigate to the VM you'd like to create a new disk for.
3. When selecting the new disk, first choose the size, of disk you need.
4. Once you've selected a size, then select a different performance tier, to change its performance.
5. Select **OK** to create the disk.

Size	Disk tier	Provisioned IOPS	Provisioned throughput	Max Shares	Max burst IOPS	Max burst throughput
4 GiB	P1	120	25	-	3500	170
8 GiB	P2	120	25	-	3500	170
16 GiB	P3	120	25	-	3500	170
32 GiB	P4	120	25	-	3500	170
64 GiB	P6	240	50	-	3500	170
128 GiB	P10	500	100	-	3500	170
256 GiB	P15	1100	125	2	3500	170
512 GiB	P20	2300	150	2	3500	170
1024 GiB	P30	5000	200	5	-	-
2048 GiB	P40	7500	250	5	-	-
4096 GiB	P50	7500	250	5	-	-
8192 GiB	P60	16000	500	10	-	-
16384 GiB	P70	18000	750	10	-	-
32767 GiB	P80	20000	900	10	-	-

## Change the performance tier of an existing disk

A disk's performance tier can be changed without downtime, so you don't have to deallocate your VM or detach your disk to change the tier.

### Change performance tier

1. Navigate to the VM containing the disk you'd like to change.
2. Select your disk
3. Select **Size + Performance**.
4. In the **Performance tier** dropdown, select a tier other than the disk's current performance tier.
5. Select **Resize**.

 **performance-tier-upgrade** | Size + performance 

Disk

Search (Ctrl+/)

Premium SSD

Size	Disk tier	Provisioned IOPS
4 GiB	P1	120
8 GiB	P2	120
16 GiB	P3	120
32 GiB	P4	120
64 GiB	P6	240
128 GiB	P10	500
256 GiB	P15	1100
512 GiB	P20	2300
1024 GiB	P30	5000
2048 GiB	P40	7500
4096 GiB	P50	7500
8192 GiB	P60	16000
16384 GiB	P70	18000
32767 GiB	P80	20000

Encryption 

Networking 

Disk Export 

Properties 

Locks 

Monitoring

Metrics 

Automation

Tasks (preview) 

Export template 

Support + troubleshooting

New support request 

Custom disk size (GiB) \* 

Performance tier 

## Next steps

If you need to resize a disk to take advantage of the higher performance tiers, see these articles:

- [Expand virtual hard disks on a Linux VM with the Azure CLI](#)
- [Expand a managed disk attached to a Windows virtual machine](#)

# Enable Write Accelerator

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Write Accelerator is a disk capability for M-Series Virtual Machines (VMs) on Premium Storage with Azure Managed Disks exclusively. As the name states, the purpose of the functionality is to improve the I/O latency of writes against Azure Premium Storage. Write Accelerator is ideally suited where log file updates are required to persist to disk in a highly performant manner for modern databases.

Write Accelerator is generally available for M-series VMs in the Public Cloud.

## Planning for using Write Accelerator

Write Accelerator should be used for the volumes that contain the transaction log or redo logs of a DBMS. It is not recommended to use Write Accelerator for the data volumes of a DBMS as the feature has been optimized to be used against log disks.

Write Accelerator only works in conjunction with [Azure managed disks](#).

### IMPORTANT

Enabling Write Accelerator for the operating system disk of the VM will reboot the VM.

To enable Write Accelerator to an existing Azure disk that is NOT part of a volume build out of multiple disks with Windows disk or volume managers, Windows Storage Spaces, Windows Scale-out file server (SOFS), Linux LVM, or MDADM, the workload accessing the Azure disk needs to be shut down. Database applications using the Azure disk MUST be shut down.

If you want to enable or disable Write Accelerator for an existing volume that is built out of multiple Azure Premium Storage disks and striped using Windows disk or volume managers, Windows Storage Spaces, Windows Scale-out file server (SOFS), Linux LVM or MDADM, all disks building the volume must be enabled or disabled for Write Accelerator in separate steps. Before enabling or disabling Write Accelerator in such a configuration, shut down the Azure VM.

Enabling Write Accelerator for OS disks should not be necessary for SAP-related VM configurations.

### Restrictions when using Write Accelerator

When using Write Accelerator for an Azure disk/VHD, these restrictions apply:

- The Premium disk caching must be set to 'None' or 'Read Only'. All other caching modes are not supported.
- Snapshots are currently supported for only Write Accelerator-enabled data disks, and not the OS disk.  
During backup, the Azure Backup service automatically backs up and protects Write Accelerator-enabled data disks attached to the VM.
- Only smaller I/O sizes (<=64 KiB) are taking the accelerated path. In workload situations where data is getting bulk loaded or where the transaction log buffers of the different DBMS are filled to a larger degree before getting persisted to the storage, chances are that the I/O written to disk is not taking the accelerated path.

There are limits of Azure Premium Storage VHDs per VM that can be supported by Write Accelerator. The current limits are:

VM SKU	NUMBER OF WRITE ACCELERATOR DISKS	WRITE ACCELERATOR DISK IOPS PER VM
M416ms_v2, M416s_v2	16	20000
M208ms_v2, M208s_v2	8	10000
M192ids_v2, M192idms_v2, M192is_v2, M192ims_v2,	16	20000
M128ms, M128s, M128ds_v2, M128dms_v2, M128s_v2, M128ms_v2	16	20000
M64ms, M64ls, M64s, M64ds_v2, M64dms_v2, M64s_v2, M64ms_v2	8	10000
M32ms, M32ls, M32ts, M32s, M32dms_v2, M32ms_v2	4	5000
M16ms, M16s	2	2500
M8ms, M8s	1	1250

The IOPS limits are per VM and *not* per disk. All Write Accelerator disks share the same IOPS limit per VM. Attached disks cannot exceed the write accelerator IOPS limit for a VM. For an example, even though the attached disks can do 30,000 IOPS, the system does not allow the disks to go above 20,000 IOPS for M416ms\_v2.

## Enabling Write Accelerator on a specific disk

The next few sections will describe how Write Accelerator can be enabled on Azure Premium Storage VHDs.

### Prerequisites

The following prerequisites apply to the usage of Write Accelerator at this point in time:

- The disks you want to apply Azure Write Accelerator against need to be [Azure managed disks](#) on Premium Storage.
- You must be using an M-series VM

## Enabling Azure Write Accelerator using Azure PowerShell

The Azure Power Shell module from version 5.5.0 include the changes to the relevant cmdlets to enable or disable Write Accelerator for specific Azure Premium Storage disks. In order to enable or deploy disks supported by Write Accelerator, the following Power Shell commands got changed, and extended to accept a parameter for Write Accelerator.

A new switch parameter, **-WriteAccelerator** has been added to the following cmdlets:

- [Set-AzVMOSDisk](#)
- [Add-AzVMDataDisk](#)
- [Set-AzVMDataDisk](#)
- [Add-AzVmssDataDisk](#)

Not giving the parameter sets the property to false and will deploy disks that have no support by Write Accelerator.

A new switch parameter, **-OsDiskWriteAccelerator** was added to the following cmdlets:

- [Set-AzVmssStorageProfile](#)

Not specifying the parameter sets the property to false by default, returning disks that don't leverage Write Accelerator.

A new optional Boolean (non-nullable) parameter, **-OsDiskWriteAccelerator** was added to the following cmdlets:

- [Update-AzVM](#)
- [Update-AzVmss](#)

Specify either \$true or \$false to control support of Azure Write Accelerator with the disks.

Examples of commands could look like:

```
New-AzVMConfig | Set-AzVMOsDisk | Add-AzVMDataDisk -Name "datadisk1" | Add-AzVMDataDisk -Name "logdisk1" -WriteAccelerator | New-AzVM

Get-AzVM | Update-AzVM -OsDiskWriteAccelerator $true

New-AzVmssConfig | Set-AzVmssStorageProfile -OsDiskWriteAccelerator | Add-AzVmssDataDisk -Name "datadisk1" -WriteAccelerator:$false | Add-AzVmssDataDisk -Name "logdisk1" -WriteAccelerator | New-AzVmss

Get-AzVmss | Update-AzVmss -OsDiskWriteAccelerator:$false
```

Two main scenarios can be scripted as shown in the following sections.

### Adding a new disk supported by Write Accelerator using PowerShell

You can use this script to add a new disk to your VM. The disk created with this script uses Write Accelerator.

Replace `myVM`, `myWAVMs`, `log001`, size of the disk, and LunID of the disk with values appropriate for your specific deployment.

```
# Specify your VM Name
$vmName="myVM"
#Specify your Resource Group
$rgName = "myWAVMs"
#data disk name
$datadiskname = "log001"
#LUN Id
$lunid=8
#size
$size=1023
#Pulls the VM info for later
$vm=Get-AzVM -ResourceGroupName $rgname -Name $vmname
#add a new VM data disk
Add-AzVMDataDisk -CreateOption empty -DiskSizeInGB $size -Name $vmname-$datadiskname -VM $vm -Caching None -WriteAccelerator:$true -lun $lunid
#Updates the VM with the disk config - does not require a reboot
Update-AzVM -ResourceGroupName $rgname -VM $vm
```

### Enabling Write Accelerator on an existing Azure disk using PowerShell

You can use this script to enable Write Accelerator on an existing disk. Replace `myVM`, `myWAVMs`, and `test-log001` with values appropriate for your specific deployment. The script adds Write Accelerator to an existing disk where the value for `$newstatus` is set to '\$true'. Using the value '\$false' will disable Write Accelerator on a given disk.

```

#Specify your VM Name
$vmName="myVM"
#Specify your Resource Group
$rgName = "myWAVMs"
#data disk name
$datadiskname = "test-log001"
#new Write Accelerator status ($true for enabled, $false for disabled)
$newstatus = $true
#Pulls the VM info for later
$vm=Get-AzVM -ResourceGroupName $rgname -Name $vmname
#add a new VM data disk
Set-AzVMDataDisk -VM $vm -Name $datadiskname -Caching None -WriteAccelerator:$newstatus
#Updates the VM with the disk config - does not require a reboot
Update-AzVM -ResourceGroupName $rgname -VM $vm

```

#### NOTE

Executing the script above will detach the disk specified, enable Write Accelerator against the disk, and then attach the disk again

## Enabling Write Accelerator using the Azure portal

You can enable Write Accelerator via the portal where you specify your disk caching settings:

LUN	NAME	SIZE	STORAGE ACCOUNT TYPE	ENCRYPTION	HOST CACHING
0	WADisk1	1023 GiB	Premium_LRS	Not enabled	Read-only + Write Accelerator
1	WADisk2	1023 GiB	Premium_LRS	Not enabled	None + Write Accelerator

## Enabling Write Accelerator using the Azure CLI

You can use the [Azure CLI](#) to enable Write Accelerator.

To enable Write Accelerator on an existing disk, use [az vm update](#), you may use the following examples if you replace the `diskName`, `VMName`, and `ResourceGroup` with your own values:

```
az vm update -g group1 -n vm1 -write-accelerator 1=true
```

To attach a disk with Write Accelerator enabled use [az vm disk attach](#), you may use the following example if you substitute in your own values: `az vm disk attach -g group1 -vm-name vm1 -disk d1 --enable-write-accelerator`

To disable Write Accelerator, use [az vm update](#), setting the properties to false:

```
az vm update -g group1 -n vm1 -write-accelerator 0=false 1=false
```

# Enabling Write Accelerator using REST APIs

To deploy through Azure REST API, you need to install the Azure armclient.

## Install armclient

To run armclient, you need to install it through Chocolatey. You can install it through cmd.exe or PowerShell. Use elevated rights for these commands ("Run as Administrator").

Using cmd.exe, run the following command:

```
@%SystemRoot%\System32\WindowsPowerShell\v1.0\powershell.exe" -NoProfile -InputFormat None -ExecutionPolicy Bypass -Command "iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))" && SET "PATH=%PATH%;%ALLUSERSPROFILE%\chocolatey\bin"
```

Using Power Shell, run the following command:

```
Set-ExecutionPolicy Bypass -Scope Process -Force; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

Now you can install the armclient by using the following command in either cmd.exe or PowerShell

```
choco install armclient
```

## Getting your current VM configuration

To change the attributes of your disk configuration, you first need to get the current configuration in a JSON file. You can get the current configuration by executing the following command:

```
armclient GET /subscriptions/<<subscription-ID>>/resourceGroups/<<ResourceGroup>>/providers/Microsoft.Compute/virtualMachines/<<virtualmachinename>>?api-version=2017-12-01 > <<filename.json>>
```

Replace the terms within '<< >>' with your data, including the file name the JSON file should have.

The output could look like:

```
{
  "properties": {
    "vmId": "2444c93e-f8bb-4a20-af2d-1658d9dbbbcb",
    "hardwareProfile": {
      "vmSize": "Standard_M64s"
    },
    "storageProfile": {
      "imageReference": {
        "publisher": "SUSE",
        "offer": "SLES-SAP",
        "sku": "12-SP3",
        "version": "latest"
      },
      "osDisk": {
        "osType": "Linux",
        "name": "mylittlesap_OsDisk_1_754a1b8bb390468e9b4c429b81cc5f5a",
        "createOption": "FromImage",
        "caching": "ReadWrite",
        "managedDisk": {
          "storageAccountType": "Premium_LRS",
          "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/mylittlesap_OsDisk_1_754a1b8bb390468e9b4c429b81cc5f5a"
        },
        "diskSizeGB": 30
      },
      "dataDisks": [
        {
          "lun": 0,
          "name": "data1",
          "createOption": "Attach",
          "caching": "None",
          "managedDisk": {
            "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data1"
          }
        }
      ]
  }
}
```

```

        "storageAccountType": "Premium_LRS",
        "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data1"
    },
    "diskSizeGB": 1023
},
{
    "lun": 1,
    "name": "log1",
    "createOption": "Attach",
    "caching": "None",
    "managedDisk": {
        "storageAccountType": "Premium_LRS",
        "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data2"
    },
    "diskSizeGB": 1023
}
],
},
"osProfile": {
    "computerName": "mylittlesapVM",
    "adminUsername": "pl",
    "linuxConfiguration": {
        "disablePasswordAuthentication": false
    },
    "secrets": []
},
"networkProfile": {
    "networkInterfaces": [
        {
            "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Network/networkInterfaces/mylittlesap518"
        }
    ]
},
"diagnosticsProfile": {
    "bootDiagnostics": {
        "enabled": true,
        "storageUri": "https://mylittlesapdiag895.blob.core.windows.net/"
    }
},
"provisioningState": "Succeeded"
},
"type": "Microsoft.Compute/virtualMachines",
"location": "westeurope",
"id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/virtualMachines/mylittlesapVM",
"name": "mylittlesapVM"

```

Next, update the JSON file and to enable Write Accelerator on the disk called 'log1'. This can be accomplished by adding this attribute into the JSON file after the cache entry of the disk.

```
{
    "lun": 1,
    "name": "log1",
    "createOption": "Attach",
    "caching": "None",
    "writeAcceleratorEnabled": true,
    "managedDisk": {
        "storageAccountType": "Premium_LRS",
        "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data2"
    },
    "diskSizeGB": 1023
}
```

Then update the existing deployment with this command:

```
armclient PUT /subscriptions/<<subscription-ID>>/resourceGroups/<<ResourceGroup>>/providers/Microsoft.Compute/virtualMachines/<<virtualmachinename>>?api-version=2017-12-01 @<<filename.json>>
```

The output should look like the one below. You can see that Write Accelerator enabled for one disk.

```
{
    "properties": {
        "vmId": "2444c93e-f8bb-4a20-af2d-1658d9dbbbc",
        "hardwareProfile": {
            "vmSize": "Standard_M64s"
        },
        "storageProfile": {
            "imageReference": {
                "publisher": "SUSE",
                "offer": "SLES-SAP",
                "sku": "12-SP3",
                "version": "latest"
            },
            "osDisk": {
                "osType": "Linux",
                "name": "mylittlesap_OsDisk_1_754a1b8bb390468e9b4c429b81cc5f5a",
                "createOption": "FromImage",
                "caching": "ReadWrite",
                "managedDisk": {
                    "storageAccountType": "Premium_LRS",
                    "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/mylittlesap_OsDisk_1_754a1b8bb390468e9b4c429b81cc5f5a"
                },
                "diskSizeGB": 30
            },
            "dataDisks": [
                {
                    "lun": 0,
                    "name": "data1",
                    "createOption": "Attach",
                    "caching": "None",
                    "managedDisk": {
                        "storageAccountType": "Premium_LRS",
                        "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data1"
                    },
                    "diskSizeGB": 1023
                },
                {
                    "lun": 1,
                    "name": "log1",
                    "createOption": "Attach",
                    "caching": "None"
                }
            ]
        }
    }
}
```

```

    "caching": "None",
    "writeAcceleratorEnabled": true,
    "managedDisk": {
        "storageAccountType": "Premium_LRS",
        "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/disks/data2"
    },
    "diskSizeGB": 1023
}
],
},
"osProfile": {
    "computerName": "mylittlesapVM",
    "adminUsername": "pl",
    "linuxConfiguration": {
        "disablePasswordAuthentication": false
    },
    "secrets": []
},
"networkProfile": {
    "networkInterfaces": [
        {
            "id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Network/networkInterfaces/mylittlesap518"
        }
    ]
},
"diagnosticsProfile": {
    "bootDiagnostics": {
        "enabled": true,
        "storageUri": "https://mylittlesapdiag895.blob.core.windows.net/"
    }
},
"provisioningState": "Succeeded"
},
"type": "Microsoft.Compute/virtualMachines",
"location": "westeurope",
"id": "/subscriptions/XXXXXXXXXXXXXXXXXXXXXX/resourceGroups/mylittlesap/providers/Microsoft.Compute/virtualMachines/mylittlesapVM",
"name": "mylittlesapVM"

```

Once you've made this change, the drive should be supported by Write Accelerator.

# Benchmark a disk

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Benchmarking is the process of simulating different workloads on your application and measuring the application performance for each workload. Using the steps described in the [designing for high performance article](#), you have gathered the application performance requirements. By running benchmarking tools on the VMs hosting the application, you can determine the performance levels that your application can achieve with premium SSDs. In this article, we provide you examples of benchmarking a Standard\_D8ds\_v4 VM provisioned with Azure premium SSDs.

We have used common benchmarking tools DiskSpd and FIO, for Windows and Linux respectively. These tools spawn multiple threads simulating a production like workload, and measure the system performance. Using the tools you can also configure parameters like block size and queue depth, which you normally cannot change for an application. This gives you more flexibility to drive the maximum performance on a high scale VM provisioned with premium SSDs for different types of application workloads. To learn more about each benchmarking tool visit [DiskSpd](#) and [FIO](#).

To follow the examples below, create a Standard\_D8ds\_v4 and attach four premium SSDs to the VM. Of the four disks, configure three with host caching as "None" and stripe them into a volume called NoCacheWrites. Configure host caching as "ReadOnly" on the remaining disk and create a volume called CacheReads with this disk. Using this setup, you are able to see the maximum Read and Write performance from a Standard\_D8ds\_v4 VM. For detailed steps about creating a Standard\_D8ds\_v4 with premium SSDs, see [Designing for high performance](#).

## Warm up the Cache

The disk with ReadOnly host caching is able to give higher IOPS than the disk limit. To get this maximum read performance from the host cache, first you must warm up the cache of this disk. This ensures that the Read IOs that the benchmarking tool will drive on CacheReads volume, actually hits the cache, and not the disk directly. The cache hits result in more IOPS from the single cache enabled disk.

### IMPORTANT

You must warm up the cache before running benchmarks every time VM is rebooted.

## DISKSPD

[Download the DISKSPD tool](#) on the VM. DISKSPD is a tool that you can customize to create your own synthetic workloads. We will use the same setup described above to run benchmarking tests. You can change the specifications to test different workloads.

In this example, we use the following set of baseline parameters:

- -c200G: Creates (or recreates) the sample file used in the test. It can be set in bytes, KiB, MiB, GiB, or blocks. In this case, a large file of 200-GiB target file is used to minimize memory caching.
- -w100: Specifies the percentage of operations that are write requests (-w0 is equivalent to 100% read).
- -b4K: Indicates the block size in bytes, KiB, MiB, or GiB. In this case, 4K block size is used to simulate a random I/O test.

- -F4: Sets a total of four threads.
- -r: Indicates the random I/O test (overrides the -s parameter).
- -o128: Indicates the number of outstanding I/O requests per target per thread. This is also known as the queue depth. In this case, 128 is used to stress the CPU.
- -W7200: Specifies the duration of the warm-up time before measurements start.
- -d30: Specifies the duration of the test, not including warm-up.
- -Sh: Disable software and hardware write caching (equivalent to -Suw).

For a complete list of parameters, see the [GitHub repository](#).

## Maximum write IOPS

We use a high queue depth of 128, a small block size of 8 KB, and four worker threads for driving Write operations. The write workers are driving traffic on the “NoCacheWrites” volume, which has three disks with cache set to “None”.

Run the following command for 30 seconds of warm-up and 30 seconds of measurement:

```
diskspd -c200G -w100 -b8K -F4 -r -o128 -W30 -d30 -Sh testfile.dat
```

Results show that the Standard\_D8ds\_v4 VM is delivering its maximum write IOPS limit of 12,800.

Total IO						
thread	bytes	I/Os	MiB/s	I/O per s	file	
0	802029568	97904	25.49	3262.63	G:\testfile.dat (200GiB)	
1	801382400	97825	25.47	3260.00	G:\testfile.dat (200GiB)	
2	802217984	97927	25.50	3263.39	G:\testfile.dat (200GiB)	
3	803012608	98024	25.52	3266.63	G:\testfile.dat (200GiB)	
total:	3208642560	391680	101.97	13052.65		
Read IO						
thread	bytes	I/Os	MiB/s	I/O per s	file	
0	0	0	0.00	0.00	G:\testfile.dat (200GiB)	
1	0	0	0.00	0.00	G:\testfile.dat (200GiB)	
2	0	0	0.00	0.00	G:\testfile.dat (200GiB)	
3	0	0	0.00	0.00	G:\testfile.dat (200GiB)	
total:	0	0	0.00	0.00		
Write IO						
thread	bytes	I/Os	MiB/s	I/O per s	file	
0	802029568	97904	25.49	3262.63	G:\testfile.dat (200GiB)	
1	801382400	97825	25.47	3260.00	G:\testfile.dat (200GiB)	
2	802217984	97927	25.50	3263.39	G:\testfile.dat (200GiB)	
3	803012608	98024	25.52	3266.63	G:\testfile.dat (200GiB)	
total:	3208642560	391680	101.97	13052.65		

## Maximum read IOPS

We use a high queue depth of 128, a small block size of four KB, and four worker threads for driving Read operations. The read workers are driving traffic on the “CacheReads” volume, which has one disk with cache set to “ReadOnly”.

Run the following command for two hours of warm-up and 30 seconds of measurement:

```
diskspd -c200G -b4K -F4 -r -o128 -W7200 -d30 -Sh testfile.dat
```

Results show that the Standard\_D8ds\_v4 VM is delivering its maximum read IOPS limit of 77,000.

Total IO	thread	bytes	I/Os	MiB/s	I/O per s	file
	0	2632953856	642811	83.66	21417.94	F:\testfile.dat (200GiB)
	1	2468286464	602609	78.43	20078.44	F:\testfile.dat (200GiB)
	2	2044755968	499208	64.97	16633.21	F:\testfile.dat (200GiB)
	3	2506788864	612009	79.65	20391.64	F:\testfile.dat (200GiB)
	total:	9652785152	2356637	306.72	78521.23	
Read IO	thread	bytes	I/Os	MiB/s	I/O per s	file
	0	2632953856	642811	83.66	21417.94	F:\testfile.dat (200GiB)
	1	2468286464	602609	78.43	20078.44	F:\testfile.dat (200GiB)
	2	2044755968	499208	64.97	16633.21	F:\testfile.dat (200GiB)
	3	2506788864	612009	79.65	20391.64	F:\testfile.dat (200GiB)
	total:	9652785152	2356637	306.72	78521.23	
Write IO	thread	bytes	I/Os	MiB/s	I/O per s	file
	0	0	0	0.00	0.00	F:\testfile.dat (200GiB)
	1	0	0	0.00	0.00	F:\testfile.dat (200GiB)
	2	0	0	0.00	0.00	F:\testfile.dat (200GiB)
	3	0	0	0.00	0.00	F:\testfile.dat (200GiB)
	total:	0	0	0.00	0.00	

### Maximum throughput

To get the maximum read and write throughput, you can change to a larger block size of 64 KB.

## FIO

FIO is a popular tool to benchmark storage on the Linux VMs. It has the flexibility to select different IO sizes, sequential or random reads and writes. It spawns worker threads or processes to perform the specified I/O operations. You can specify the type of I/O operations each worker thread must perform using job files. We created one job file per scenario illustrated in the examples below. You can change the specifications in these job files to benchmark different workloads running on Premium Storage. In the examples, we are using a Standard\_D8ds\_v4 running **Ubuntu**. Use the same setup described in the beginning of the benchmark section and warm up the cache before running the benchmark tests.

Before you begin, [download FIO](#) and install it on your virtual machine.

Run the following command for Ubuntu,

```
apt-get install fio
```

We use four worker threads for driving Write operations and four worker threads for driving Read operations on the disks. The write workers are driving traffic on the "nocache" volume, which has three disks with cache set to "None". The read workers are driving traffic on the "readcache" volume, which has one disk with cache set to "ReadOnly".

### Maximum write IOPS

Create the job file with following specifications to get maximum Write IOPS. Name it "fiowrite.ini".

```

[global]
size=30g
direct=1
iodepth=256
ioengine=libaio
bs=4k
numjobs=4

[writer1]
rw=randwrite
directory=/mnt/nocache

```

Note the follow key things that are in line with the design guidelines discussed in previous sections. These specifications are essential to drive maximum IOPS,

- A high queue depth of 256.
- A small block size of 4 KB.
- Multiple threads performing random writes.

Run the following command to kick off the FIO test for 30 seconds,

```
sudo fio --runtime 30 fiowrite.ini
```

While the test runs, you are able to see the number of write IOPS the VM and Premium disks are delivering. As shown in the sample below, the Standard\_D8ds\_v4 VM is delivering its maximum write IOPS limit of 12,800 IOPS.

```

:~$ sudo fio fiowriteuncached.ini
writer1: (g=0): rw=randwrite, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=256
...
fio-3.1
Starting 4 processes
Jobs: 4 (f=4): [w(4)][0.2%][r=0KiB/s,w=51.1MiB/s][r=0, w=13.1k IOPS][eta 54m:23s]

```

## Maximum read IOPS

Create the job file with following specifications to get maximum Read IOPS. Name it "fioread.ini".

```

[global]
size=30g
direct=1
iodepth=256
ioengine=libaio
bs=4k
numjobs=4

[reader1]
rw=randread
directory=/mnt/readcache

```

Note the follow key things that are in line with the design guidelines discussed in previous sections. These specifications are essential to drive maximum IOPS,

- A high queue depth of 256.
- A small block size of 4 KB.
- Multiple threads performing random writes.

Run the following command to kick off the FIO test for 30 seconds,

```
sudo fio --runtime 30 fioread.ini
```

While the test runs, you are able to see the number of read IOPS the VM and Premium disks are delivering. As shown in the sample below, the Standard\_D8ds\_v4 VM is delivering more than 77,000 Read IOPS. This is a combination of the disk and the cache performance.

```
:~$ sudo fio fioreadcache.ini
reader1: (g=0): rw=randread, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=256
...
fio-3.1
Starting 4 processes
Jobs: 4 (f=4): [r(4)][2.0%][r=307MiB/s,w=0KiB/s] [r=78.6k,w=0 IOPS][eta 08m:59s]
```

## Maximum read and write IOPS

Create the job file with following specifications to get maximum combined Read and Write IOPS. Name it "fioreadwrite.ini".

```
[global]
size=30g
direct=1
iodepth=128
ioengine=libaio
bs=4k
numjobs=4

[reader1]
rw=randread
directory=/mnt/readcache

[writer1]
rw=randwrite
directory=/mnt/nocache
rate_iops=3200
```

Note the follow key things that are in line with the design guidelines discussed in previous sections. These specifications are essential to drive maximum IOPS,

- A high queue depth of 128.
- A small block size of 4 KB.
- Multiple threads performing random reads and writes.

Run the following command to kick off the FIO test for 30 seconds,

```
sudo fio --runtime 30 fioreadwrite.ini
```

While the test runs, you are able to see the number of combined read and write IOPS the VM and Premium disks are delivering. As shown in the sample below, the Standard\_D8ds\_v4 VM is delivering more than 90,000 combined Read and Write IOPS. This is a combination of the disk and the cache performance.

```
:~$ sudo fio fioboth.ini
reader1: (g=0): rw=randread, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=256
...
writer1: (g=0): rw=randwrite, bs=(R) 4096B-4096B, (W) 4096B-4096B, (T) 4096B-4096B, ioengine=libaio, iodepth=256
...
fio-3.1
Starting 8 processes
Jobs: 8 (f=8), 0-12800 IOPS: [r(4),w(4)][0.7%][r=306MiB/s,w=49.4MiB/s] [r=78.3k,w=12.6k IOPS][eta 54m:36s]]
```

## Maximum combined throughput

To get the maximum combined Read and Write Throughput, use a larger block size and large queue depth with multiple threads performing reads and writes. You can use a block size of 64 KB and queue depth of 128.

## Next steps

Proceed to our article on [designing for high performance](#).

In that article, you create a checklist similar to your existing application for the prototype. Using Benchmarking tools you can simulate the workloads and measure performance on the prototype application. By doing so, you can determine which disk offering can match or surpass your application performance requirements. Then you can implement the same guidelines for your production application.

# Scalability and performance targets for VM disks

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can attach a number of data disks to an Azure virtual machine (VM). Based on the scalability and performance targets for a VM's data disks, you can determine the number and type of disk that you need to meet your performance and capacity requirements.

## IMPORTANT

For optimal performance, limit the number of highly utilized disks attached to the virtual machine to avoid possible throttling. If all attached disks aren't highly utilized at the same time, the virtual machine can support a larger number of disks.

### For Azure managed disks:

The following table illustrates the default and maximum limits of the number of resources per region per subscription. The limits remain the same irrespective of disks encrypted with either platform-managed keys or customer-managed keys. There is no limit for the number of Managed Disks, snapshots and images per resource group.

RESOURCE	LIMIT
Standard managed disks	50,000
Standard SSD managed disks	50,000
Premium managed disks	50,000
Standard_LRS snapshots <sup>1</sup>	75,000
Standard_ZRS snapshots <sup>1</sup>	75,000
Managed image	50,000

<sup>1</sup>An individual disk can have 500 incremental snapshots.

### For standard storage accounts:

A Standard storage account has a maximum total request rate of 20,000 IOPS. The total IOPS across all of your virtual machine disks in a Standard storage account should not exceed this limit.

For unmanaged disks, you can roughly calculate the number of highly utilized disks supported by a single standard storage account based on the request rate limit. For example, for a Basic tier VM, the maximum number of highly utilized disks is about 66, which is 20,000/300 IOPS per disk. The maximum number of highly utilized disks for a Standard tier VM is about 40, which is 20,000/500 IOPS per disk.

### For premium storage accounts:

A premium storage account has a maximum total throughput rate of 50 Gbps. The total throughput across all of

your VM disks should not exceed this limit.

See [VM sizes](#) for additional details.

## Managed virtual machine disks

Sizes denoted with an asterisk are currently in preview. See our [FAQ](#) to learn what regions they are available in.

### Standard HDD managed disks

STANDARD DISK TYPE	S4	S6	S10	S15	S20	S30	S40	S50	S60	S70	S80
Disk size in GiB	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
IOPS per disk	Up to 500	Up to 1,300	Up to 2,000	Up to 2,000							
Throughput per disk	Up to 60 MB/sec	Up to 300 MB/sec	Up to 500 MB/sec	Up to 500 MB/sec							

### Standard SSD managed disks

STANDARD SSD SIZES	E1	E2	E3	E4	E6	E10	E15	E20	E30	E40	E50	E60	E70	E80
Disk size in GiB	4	8	16	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
IOPS per disk	Up to 500	Up to 2,000	Up to 4,000	Up to 6,000										
Throughput per disk	Up to 60 MB/sec	Up to 400 MB/sec	Up to 600 MB/sec	Up to 750 MB/sec										

STA ND AR D SSD SIZ ES	E1	E2	E3	E4	E6	E10	E15	E20	E30	E40	E50	E60	E70	E80
Max burst IOPS per disk	600	600	600	600	600	600	600	600	1000					
Max burst throughput per disk	150 MB/sec	250 MB/sec												
Max burst duration	30 min													

#### Premium SSD managed disks: Per-disk limits

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Disk size in GiB	4	8	16	32	64	128	256	512	1,024	2,048	4,096	8,192	16,384	32,767
Provisioned IOPS per disk	120	120	120	120	240	500	1,100	2,300	5,000	7,500	7,500	16,000	18,000	20,000

PRE MIU M SSD SIZ ES	P1	P2	P3	P4	P6	P10	P15	P20	P30	P40	P50	P60	P70	P80
Pro visi one d Thr oug hpu t per disk	25 MB/ sec	25 MB/ sec	25 MB/ sec	25 MB/ sec	50 MB/ sec	100 MB/ sec	125 MB/ sec	150 MB/ sec	200 MB/ sec	250 MB/ sec	250 MB/ sec	500 MB/ sec	750 MB/ sec	900 MB/ sec
Ma x bur st IOP S per disk	3,5 00	30, 000 *	30, 000 *	30, 000 *	30, 000 *	30, 000 *	30, 000 *							
Ma x bur st thr oug hpu t per disk	170 MB/ sec	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*	1,0 00 MB/ sec*							
Ma x bur st dur atio n	30 min	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*	Unli mit ed*							
Eligi ble for rese rvat ion	No	Yes, up to one yea r	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year	Yes, up to one year							

\*Applies only to disks with on-demand bursting enabled.

#### Premium SSD managed disks: Per-VM limits

RESOURCE	LIMIT
Maximum IOPS Per VM	80,000 IOPS with GS5 VM

RESOURCE	LIMIT
Maximum throughput per VM	2,000 MB/s with GS5 VM

## Unmanaged virtual machine disks

### Standard unmanaged virtual machine disks: Per-disk limits

VM TIER	BASIC TIER VM	STANDARD TIER VM
Disk size	4,095 GB	4,095 GB
Maximum 8-KB IOPS per persistent disk	300	500
Maximum number of disks that perform the maximum IOPS	66	40

### Premium unmanaged virtual machine disks: Per-account limits

RESOURCE	LIMIT
Total disk capacity per account	35 TB
Total snapshot capacity per account	10 TB
Maximum bandwidth per account (ingress + egress) <sup>1</sup>	<=50 Gbps

<sup>1</sup>*Ingress* refers to all data from requests that are sent to a storage account. *Egress* refers to all data from responses that are received from a storage account.

### Premium unmanaged virtual machine disks: Per-disk limits

PREMIUM STORAGE DISK TYPE	P10	P20	P30	P40	P50
Disk size	128 GiB	512 GiB	1,024 GiB (1 TB)	2,048 GiB (2 TB)	4,095 GiB (4 TB)
Maximum IOPS per disk	500	2,300	5,000	7,500	7,500
Maximum throughput per disk	100 MB/sec	150 MB/sec	200 MB/sec	250 MB/sec	250 MB/sec
Maximum number of disks per storage account	280	70	35	17	8

### Premium unmanaged virtual machine disks: Per-VM limits

RESOURCE	LIMIT
Maximum IOPS per VM	80,000 IOPS with GS5 VM
Maximum throughput per VM	2,000 MB/sec with GS5 VM

## See also

[Azure subscription and service limits, quotas, and constraints](#)

# Create an incremental snapshot for managed disks

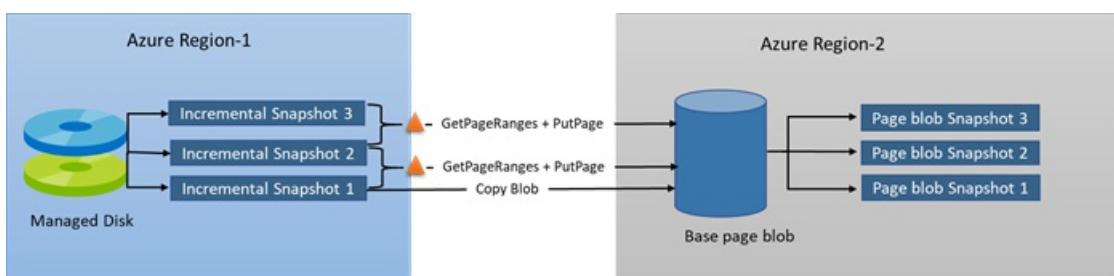
9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Incremental snapshots are point in time backups for managed disks that, when taken, consist only of the changes since the last snapshot. When you restore a disk from an incremental snapshot, the system reconstructs the full disk which represents the point in time backup of the disk when the incremental snapshot was taken. This capability for managed disk snapshots potentially allows them to be more cost effective, since, unless you choose to, you do not have to store the entire disk with each individual snapshot. Just like full snapshots, incremental snapshots can be used to either create a full managed disk or a full snapshot. Both full snapshots and incremental snapshots can be used immediately after being taken. In other words, once you take either snapshot, you can immediately read the underlying VHD and use it to restore disks.

There are a few differences between an incremental snapshot and a full snapshot. Incremental snapshots will always use standard HDD storage, irrespective of the storage type of the disk, whereas full snapshots can use premium SSDs. If you are using full snapshots on Premium Storage to scale up VM deployments, we recommend you use custom images on standard storage in the [Shared Image Gallery](#). It will help you achieve a more massive scale with lower cost. Additionally, incremental snapshots potentially offer better reliability with [zone-redundant storage](#) (ZRS). If ZRS is available in the selected region, an incremental snapshot will use ZRS automatically. If ZRS is not available in the region, then the snapshot will default to [locally-redundant storage](#) (LRS). You can override this behavior and select one manually but, we do not recommend that.

Incremental snapshots offer a differential capability. They enable you to get the changes between two incremental snapshots of the same managed disk, down to the block level. You can use this to reduce your data footprint when copying snapshots across regions. For example, you can download the first incremental snapshot as a base blob in another region. For the subsequent incremental snapshots, you can copy only the changes since the last snapshot to the base blob. After copying the changes, you can take snapshots on the base blob that represent your point in time backup of the disk in another region. You can restore your disk either from the base blob or from a snapshot on the base blob in another region.



Incremental snapshots are billed for the used size only. You can find the used size of your snapshots by looking at the [Azure usage report](#). For example, if the used data size of a snapshot is 10 GiB, the [daily usage report](#) will show  $10 \text{ GiB}/(31 \text{ days}) = 0.3226$  as the consumed quantity.

## Restrictions

- Incremental snapshots currently can't be moved between subscriptions.
- You can currently only generate SAS URLs of up to five snapshots of a particular snapshot family at any given time.
- You can't create an incremental snapshot for a particular disk outside of that disk's subscription.
- Incremental snapshots can't be moved to another resource group. But, they can be copied to another resource group or region.

- Up to seven incremental snapshots per disk can be created every five minutes.
- A total of 500 incremental snapshots can be created for a single disk.
- You can't get the changes between snapshots taken before and after you changed the size of the parent disk across 4 TB boundary. For example, You took an incremental snapshot `snapshot-a` when the size of a disk was 2 TB. Now you increased the size of the disk to 6 TB and then took another incremental snapshot `snapshot-b`. You can't get the changes between `snapshot-a` and `snapshot-b`. You have to again download the full copy of `snapshot-b` created after the resize. Subsequently, you can get the changes between `snapshot-b` and snapshots created after `snapshot-b`.

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)
- [Resource Manager Template](#)

You can use the Azure CLI to create an incremental snapshot. You will need the latest version of the Azure CLI. See the following articles to learn how to either [install](#) or [update](#) the Azure CLI.

The following script will create an incremental snapshot of a particular disk:

```
# Declare variables
diskName="yourDiskNameHere"
resourceGroupName="yourResourceGroupNameHere"
snapshotName="desiredSnapshotNameHere"

# Get the disk you need to backup
yourDiskID=$(az disk show -n $diskName -g $resourceGroupName --query "id" --output tsv)

# Create the snapshot
az snapshot create -g $resourceGroupName -n $snapshotName --source $yourDiskID --incremental true
```

You can identify incremental snapshots from the same disk with the `SourceResourceId` property of snapshots. `SourceResourceId` is the Azure Resource Manager resource ID of the parent disk.

You can use `SourceResourceId` to create a list of all snapshots associated with a particular disk. Replace `yourResourceGroupNameHere` with your value and then you can use the following example to list your existing incremental snapshots:

```
# Declare variables and create snapshot list
subscriptionId="yourSubscriptionId"
resourceGroupName="yourResourceGroupNameHere"
diskName="yourDiskNameHere"

az account set --subscription $subscriptionId

diskId=$(az disk show -n $diskName -g $resourceGroupName --query [id] -o tsv)

az snapshot list --query "[?creationData.sourceResourceId=='$diskId' && incremental]" -g $resourceGroupName
--output table
```

## Next steps

See [Copy an incremental snapshot to a new region](#) to learn how to copy an incremental snapshot across regions.

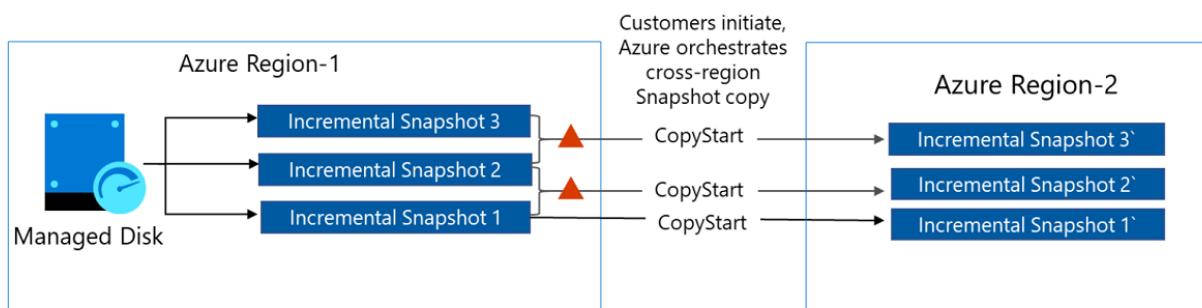
If you'd like to see sample code demonstrating the differential capability of incremental snapshots, using .NET, see [Copy Azure Managed Disks backups to another region with differential capability of incremental snapshots](#).

# Copy an incremental snapshot to a new region

9/21/2022 • 2 minutes to read • [Edit Online](#)

Incremental snapshots can be copied to any region. The process is managed by Azure, removing the maintenance overhead of managing the copy process by staging a storage account in the target region. Azure ensures that only changes since the last snapshot in the target region are copied to the target region to reduce the data footprint, reducing the recovery point objective. You can check the progress of the copy so you know when a target snapshot is ready to restore disks in the target region. You're only charged for the bandwidth cost of the data transfer across the region and the read transactions on the source snapshots.

This article covers copying an incremental snapshot from one region to another. See [Create an incremental snapshot for managed disks](#) for conceptual details on incremental snapshots.



## Restrictions

- You can copy 100 incremental snapshots in parallel at the same time per subscription per region.
- If you use the REST API, you must use version 2020-12-01 or newer of the Azure Compute REST API.

## Get started

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Portal](#)
- [Resource Manager Template](#)

You can use the Azure CLI to copy an incremental snapshot. You need the latest version of the Azure CLI. See the following articles to learn how to either [install](#) or [update](#) the Azure CLI.

The following script copies an incremental snapshot from one region to another:

```
subscriptionId=<yourSubscriptionID>
resourceGroupName=<yourResourceGroupName>
targetSnapshotName=<name>
sourceSnapshotResourceId=<sourceSnapshotResourceId>
targetRegion=<validRegion>

sourceSnapshotId=$(az snapshot show -n $sourceSnapshotName -g $resourceGroupName --query [id] -o tsv)

az snapshot create -g $resourceGroupName -n $targetSnapshotName -l $targetRegion --source $sourceSnapshotId
--incremental --copy-start

az snapshot show -n $sourceSnapshotName -g $resourceGroupName --query [completionPercent] -o tsv
```

## Next steps

If you'd like to see sample code demonstrating the differential capability of incremental snapshots, using .NET, see [Copy Azure Managed Disks backups to another region with differential capability of incremental snapshots](#).

# Overview of Azure Disk Backup

9/21/2022 • 7 minutes to read • [Edit Online](#)

Azure Disk Backup is a native, cloud-based backup solution that protects your data in managed disks. It's a simple, secure, and cost-effective solution that enables you to configure protection for managed disks in a few steps. It assures that you can recover your data in a disaster scenario.

Azure Disk Backup offers a turnkey solution that provides snapshot lifecycle management for managed disks by automating periodic creation of snapshots and retaining it for configured duration using backup policy. You can manage the disk snapshots with zero infrastructure cost and without the need for custom scripting or any management overhead. This is a crash-consistent backup solution that takes point-in-time backup of a managed disk using [incremental snapshots](#) with support for multiple backups per day. It's also an agent-less solution and doesn't impact production application performance. It supports backup and restore of both OS and data disks (including shared disks), whether or not they're currently attached to a running Azure virtual machine.

If you require application-consistent backup of virtual machine including the data disks, or an option to restore an entire virtual machine from backup, restore a file or folder, or restore to a secondary region, then use the [Azure VM backup](#) solution. Azure Backup offers side-by-side support for backup of managed disks using Disk Backup in addition to [Azure VM backup](#) solutions. This is useful when you need once-a-day application consistent backups of virtual machines and also more frequent backups of OS disks or a specific data disk that are crash consistent, and don't impact the production application performance.

Azure Disk Backup is integrated into Backup Center, which provides a **single unified management experience** in Azure for enterprises to govern, monitor, operate, and analyze backups at scale.

## Key benefits of Disk Backup

Azure Disk backup is an agentless and crash consistent solution that uses [incremental snapshots](#) and offers the following advantages:

- More frequent and quick backups without interrupting the virtual machine.
- Doesn't affect the performance of the production application.
- No security concerns since it doesn't require running custom scripts or installing agents.
- A cost-effective solution to back up specific disks when compared to backing up entire virtual machine.

Azure Disk backup solution is useful in the following scenarios:

- Need for frequent backups per day without application being quiescent.
- Apps running in a cluster scenario: both Windows Server Failover Cluster and Linux clusters are writing to shared disks.
- Specific need for agentless backup because of security or performance concerns on the application.
- Application consistent backup of VM isn't feasible since line-of-business apps don't support Volume Shadow Copy Service (VSS).

Consider Azure Disk Backup in scenarios where:

- A mission-critical application is running on an Azure Virtual machine that demands multiple backups per day to meet the recovery point objective, but without impacting the production environment or application performance.
- Your organization or industry regulation restricts installing agents because of security concerns.
- Executing custom pre or post scripts and invoking freeze and thaw on Linux virtual machines to get

application-consistent backup puts undue overhead on production workload availability.

- Containerized applications running on Azure Kubernetes Service (AKS cluster) are using managed disks as persistent storage. Today, you must back up the managed disk via automation scripts that are hard to manage.
- A managed disk is holding critical business data, used as a file-share, or contains database backup files, and you want to optimize backup cost by not investing in Azure VM backup.
- You have many Linux and Windows single-disk virtual machines (that is, a virtual machine with just an OS disk and no data disks attached) that host web server, state-less machines, or serves as a staging environment with application configuration settings, and you need a cost efficient backup solution to protect the OS disk. For example, to trigger a quick on-demand backup before upgrading or patching the virtual machine.
- A virtual machine is running an OS configuration that is unsupported by Azure VM backup solution (for example, Windows 2008 32-bit Server).

## How the backup and restore process works

- The first step in configuring backup for Azure Managed Disks is creating a [Backup vault](#). The vault gives you a consolidated view of the backups configured across different workloads. Azure Disk backup supports only Operational Tier backup. Copying of backups to the vault storage tier is not supported. So, the Backup vault storage redundancy setting (LRS/GRS) doesn't apply to the backups stored in Operational Tier.
- Then create a Backup policy that allows you to configure backup frequency and retention duration.
- To configure backup, go to the Backup vault, assign a backup policy, select the managed disk that needs to be backed up and provide a resource group where the snapshots are to be stored and managed. Azure Backup automatically triggers scheduled backup jobs that create an incremental snapshot of the disk according to the backup frequency. Older snapshots are deleted according to the retention duration specified by the backup policy.
- Azure Backup uses [incremental snapshots](#) of the managed disk. Incremental snapshots are a cost-effective, point-in-time backup of managed disks that are billed for the delta changes to the disk since the last snapshot. These are always stored on the most cost-effective storage, standard HDD storage regardless of the storage type of the parent disks. The first snapshot of the disk will occupy the used size of the disk, and consecutive incremental snapshots store delta changes to the disk since the last snapshot. Azure Backup automatically assigns tag to the snapshots it creates to uniquely identify them.
- Once you configure the backup of a managed disk, a backup instance will be created within the backup vault. Using the backup instance, you can find health of backup operations, trigger on-demand backups, and perform restore operations. You can also view health of backups across multiple vaults and backup instances using Backup Center, which provides a single pane of glass view.
- To restore, just select the recovery point from which you want to restore the disk. Provide the resource group where the restored disk is to be created from the snapshot. Azure Backup provides an instant restore experience since the snapshots are stored locally in your subscription.
- Backup Vault uses Managed Identity to access other Azure resources. To configure backup of a managed disk and to restore from past backup, Backup Vault's managed identity requires a set of permissions on the source disk, the snapshot resource group where snapshots are created and managed, and the target resource group where you want to restore the backup. You can grant permissions to the managed identity by using Azure role-based access control (Azure RBAC). Managed identity is a service principal of a special type that may only be used with Azure resources. Learn more about [Managed Identities](#).
- Currently Azure Disk Backup supports operational backup of managed disks and doesn't copy the backups to Backup Vault storage. Refer to the [support matrix](#) for a detailed list of supported and

unsupported scenarios, and region availability.

## Pricing

Azure Backup uses [incremental snapshots](#) of the managed disk. Incremental snapshots are charged per GiB of the storage occupied by the delta changes since the last snapshot. For example, if you're using a managed disk with a provisioned size of 128 GiB, with 100 GiB used, the first incremental snapshot is billed only for the used size of 100 GiB. 20 GiB of data is added on the disk before you create the second snapshot. Now, the second incremental snapshot is billed for only 20 GiB.

Incremental snapshots are always stored on standard storage, irrespective of the storage type of parent-managed disks, and are charged based on the pricing of standard storage. For example, incremental snapshots of a Premium SSD-Managed Disk are stored on standard storage. By default, they are stored on ZRS in regions that support ZRS. Otherwise, they are stored on locally redundant storage (LRS). The per GiB pricing of both the options, LRS and ZRS, is the same.

The snapshots created by Azure Backup are stored in the resource group within your Azure subscription and incur Snapshot Storage charges. For more details about the snapshot pricing, see [Managed Disk Pricing](#). Because the snapshots aren't copied to the Backup Vault, Azure Backup doesn't charge a Protected Instance fee and Backup Storage cost doesn't apply.

The number of recovery points is determined by the Backup policy used to configure backups of the disk backup instances. Older block blobs are deleted according to the garbage collection process as the corresponding older recovery points are pruned.

## Next steps

[Azure Disk Backup support matrix](#)

# Back up Azure Managed Disks

9/21/2022 • 9 minutes to read • [Edit Online](#)

This article explains how to back up [Azure Managed Disk](#) from the Azure portal.

In this article, you'll learn how to:

- Create a Backup vault
- Create a backup policy
- Configure a backup of an Azure Disk
- Run an on-demand backup job

For information on the Azure Disk backup region availability, supported scenarios and limitations, see the [support matrix](#).

## Create a Backup vault

A Backup vault is a storage entity in Azure that holds backup data for various newer workloads that Azure Backup supports, such as Azure Database for PostgreSQL servers and Azure Disks. Backup vaults make it easy to organize your backup data, while minimizing management overhead. Backup vaults are based on the Azure Resource Manager model of Azure, which provides enhanced capabilities to help secure backup data.

1. Sign in to the Azure portal at <https://portal.azure.com>.
2. Type **Backup center** in the search box.
3. Under **Services**, select **Backup center**.
4. In the **Backup center** page, select **Vault**.

5. In the **Initiate: Create Vault** screen, select **Backup vault**, and **Proceed**.

A vault is an entity that stores the backups and recovery points created over time. The vault also contains the backup policies that are associated with the protected virtual machines. Proceed to vault creation by selecting vault type.

Vault Type

Recovery Services vault

Backup vault

Supported datasources

- Azure Virtual machines
- SQL in Azure VM
- Azure Storage (Azure Files)
- SAP HANA in Azure VM
- Azure Backup Server
- Azure Backup Agent
- DPM

Learn more about Backup vault. [Click here](#)

Learn more about Recovery Services vault. [Click here](#)

**Proceed** **Cancel**

- In the **Basics** tab, provide subscription, resource group, backup vault name, region, and backup storage redundancy. Continue by selecting **Review + create**. Learn more about [creating a Backup vault](#).

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Backup center (Preview) > Initiate: Create Vault >

## Create Backup Vault

Data Protection (Preview)

**Basics** **Tags** **Review + create**

A backup vault is a storage entity in Azure that houses data and lets you organize your backups. [Learn more](#)

**PROJECT DETAILS**

Select the subscription and the resource group in which you want to create the vault.

Subscription \*

Resource Group \*  [Create new](#)

**INSTANCE DETAILS**

Backup vault name \*

Region \*

Backup storage redundancy \*

Storage redundancy cannot be changed after protecting items to the vault. Geo-redundant storage provides a higher level of data durability than Locally-redundant storage but costs more. Review the trade-offs between lower cost and higher data durability that is best for your scenario. [Learn more](#)

**Review + create** **Next: Tags >**

## Create Backup policy

- In the **DemoVault** **Backup vault** created in the previous step, go to **Backup policies** and select **Add**.

DemoVault | Backup policies

Backup vault

Search (Ctrl+)/ Add Refresh

Subscription == <subscription> Resource group == contosorg Datasource type == Azure Database for PostgreSQL

Overview Activity log Tags Filter by name

Manage

Properties Identity **Backup policies** (highlighted)

Backup instances Locks

Name ↑↓ Datasource type

No data available

< Previous 1 Next >

2. In the **Basics** tab, provide policy name, select **Datasource type** as **Azure Disk**. The vault is already prepopulated and the selected vault properties are presented.

**NOTE**

Although the selected vault may have the global-redundancy setting, currently Azure Disk Backup supports snapshot datastore only. All backups are stored in a resource group in your subscription and aren't copied to backup vault storage.

Microsoft Azure (Preview) Search resources, services, and docs (G+/)

Home > DemoVault > Create Backup Policy

Basics Backup policy Review + create

Policy name \*

Datasource type \* Azure Disk

Vault \* DemoVault

Selected Backup vault details

Subscription	<subscription>
Resource group	contosorg
Location	Southeast Asia
Backup storage redundancy	Globally-redundant

Review + create Next: Backup policy >

3. In the **Backup policy** tab, select the backup schedule frequency.

The screenshot shows the 'Create Backup Policy' page in the Microsoft Azure portal. The 'Backup policy' tab is selected. Under 'Backup schedule', 'Hourly' is chosen as the backup frequency with a time interval of 'Every 4 hours'. In the 'Retention settings' section, there is a single retention rule named 'Default' set to '7 Days'. At the bottom, there are 'Review + create', 'Previous', and 'Next: Review + create >' buttons.

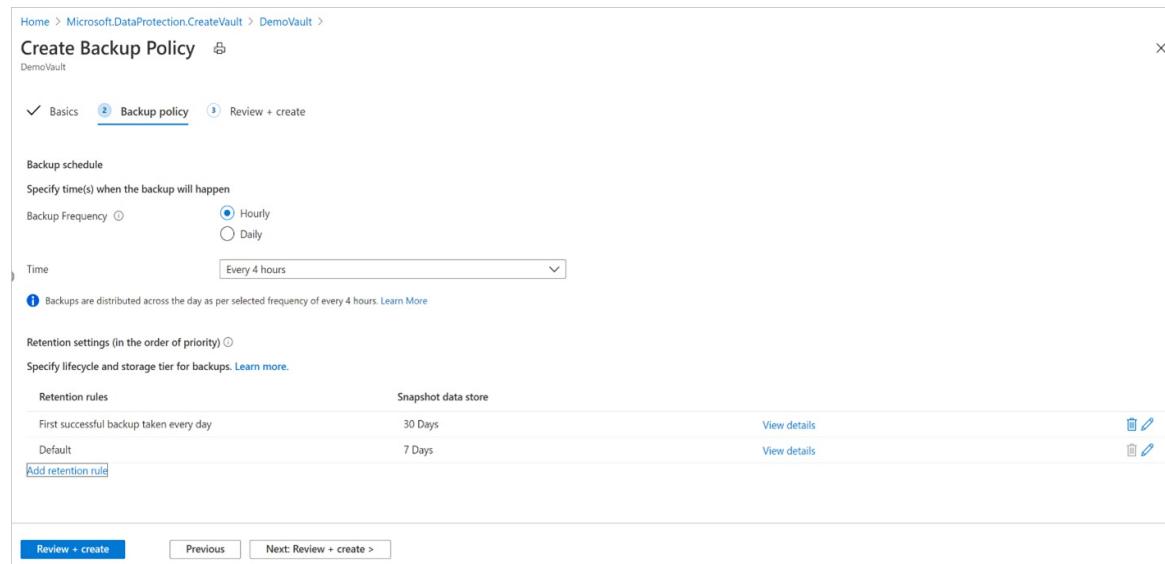
Azure Disk Backup offers multiple backups per day. If you require more frequent backups, choose the **Hourly** backup frequency with the ability to take backups with intervals of every 1, 2, 4, 6, 8, or 12 hours. The backups are scheduled based on the **Time** interval selected. For example, if you select **Every 4 hours**, then the backups are taken at approximately in the interval of every 4 hours so the backups are distributed equally across the day. If a once a day backup is sufficient, then choose the **Daily** backup frequency. In the daily backup frequency, you can specify the time of the day when your backups are taken. It's important to note that the time of the day indicates the backup start time and not the time when the backup completes. The time required for completing the backup operation is dependent on various factors including size of the disk, and churn rate between consecutive backups. However, Azure Disk backup is an agentless backup that uses [incremental snapshots](#), which doesn't impact the production application performance.

4. In the **Backup policy** tab, select retention settings that meet the recovery point objective (RPO) requirement.

The default retention rule applies if no other retention rule is specified. The default retention rule can be modified to change the retention duration, but it cannot be deleted. You can add a new retention rule by selecting **Add retention rule**.

The screenshot shows the 'Add retention' dialog box overlaid on the 'Create Backup Policy' page. The dialog box contains fields for 'Retention rules' (set to 'Backup taken every day') and 'Data store' (set to 'Snapshot data store' with a duration of '30 Day(s)'). A 'Cancel' button is located at the bottom right of the dialog.

You can pick first successful backup taken daily or weekly, and provide the retention duration that the specific backups are to be retained before they're deleted. This option is useful to retain specific backups of the day or week for a longer duration of time. All other frequent backups can be retained for a shorter duration.



The screenshot shows the 'Create Backup Policy' wizard in progress. The current step is 'Backup policy'. The 'Backup schedule' section shows 'Backup Frequency' set to 'Hourly' with 'Every 4 hours' selected. The 'Retention settings (in the order of priority)' section lists two rules: 'First successful backup taken every day' with a 'Snapshot data store' of '30 Days', and a 'Default' rule with a 'Snapshot data store' of '7 Days'. At the bottom, there are buttons for 'Review + create', 'Previous', and 'Next: Review + create >'.

#### NOTE

Azure Backup for Managed Disks uses incremental snapshots which are limited to 200 snapshots per disk. To allow you to take on-demand backups aside from scheduled backups, backup policy limits the total backups to 180. Learn more about [incremental snapshots for managed disk](#).

5. Complete the backup policy creation by selecting **Review + create**.

## Configure backup

- Azure Disk backup supports only the operational tier backup. Copying of backups to the vault storage tier is currently not supported. The Backup vault storage redundancy setting (LRS/GRS) doesn't apply to the backups stored in the operational tier.  
Incremental snapshots are stored in a Standard HDD storage, irrespective of the selected storage type of the parent disk. For additional reliability, incremental snapshots are stored on **Zone Redundant Storage (ZRS)** by default in ZRS supported regions.
- Azure Disk backup supports cross-subscription (backup vault in one subscription and the source disk in another) backup and restore. Currently, cross-region backup and restore aren't supported by Azure Disk backup, that is, the backup vault and disk to back up are in different regions.  
So, to use Azure Disk backup, ensure that the backup vault and disk to back up are in the same region.
- Once you configure the disk backup, you can't change the Snapshot Resource Group that's assigned to a backup instance.

To configure disk backup, follow these steps:

1. Go to **Backup center** -> **Overview** and click **+ Backup** to start configuring backup of the disk.

The screenshot shows the Microsoft Azure Backup center interface. On the left, there's a navigation sidebar with sections like Overview, Getting started, Community, Manage (Backup instances, Backup policies, Vaults, Monitoring + reporting, Backup jobs, Alerts, Backup reports), Policy and compliance (Backup compliance, Azure policies for backup, Protectable datasources), and Support + troubleshooting (New support request). The main area displays the following information:

- Datasource type: Azure Disks**
- Jobs (last 24 Hours)**: Shows 97 Failed, 5 In progress, and 436 Completed scheduled backups.
- Backup instances**: Shows 122 total instances, with 99 Protection configured, 16 Protection error, and 0 Configuring protection failed. It also indicates 10 instances out of 122 have underlying datasource issues.
- Active Alerts (last 24 Hours)**: Shows 0 Security alerts and 98 Configured alerts. A legend indicates 0 Sev0, 98/98 Sev1, and 0 Sev2.
- Global alerts**: Shows 140 Security alerts and 0 Configured alerts. A legend indicates 140/140 Sev0, 0/140 Sev1, and 0/140 Sev2.

2. Select Azure Disks in the Datasource type drop-down list, and then click Continue.

The screenshot shows the 'Start: Configure Backup' step of the Azure Backup wizard. It has two dropdown menus: 'Datasource type' set to 'Azure Disks' and 'Vault type' set to 'Backup vault'. At the bottom are 'Continue' and 'Cancel' buttons.

3. Select a Backup vault and click Next to proceed.

#### NOTE

- Ensure that both the backup vault and the disk to be backed up are in same location.
- Azure Backup uses *incremental snapshots* of managed disks, which store only the delta changes to the disk as the last snapshot on Standard HDD storage, regardless of the storage type of the parent disk. For additional reliability, incremental snapshots are stored on Zone Redundant Storage (ZRS) by default in the ZRS supported regions. Currently, Azure Disk Backup supports operational backup of managed disks that doesn't copy backups to the Backup vault storage. So, the backup storage redundancy setting of the Backup vault doesn't apply to the recovery points.

Microsoft Azure (Preview) Search resources, services, and docs (G+/-)

Home > Backup center > Start: Configure Backup >

Configure Backup ...

Basics Backup policy Datasources Review and configure

Datasource type \* Azure Disks

Use a backup for an Az Assign backup policy to regardless of whether t provides agentless bad operational data store.

Vault \* Select a Vault

Select Create new

The backup storage redundancy setting doesn't apply to backups st the operational data store use Standard HDD storage and zone-redu

Next Select Cancel

Resource type: **Backup vault**

Subscription == 48 selected Location == All Resource group == 2 selected

1-1 of 1 items

Name	Subscription	Resource group
DemoVault	<subscription>	<resource group>

< Previous 1 Next >

4. On the **Backup Policy** tab, choose a Backup policy.

Microsoft Azure (Preview) Search resources, services, and docs (G+/-)

Home > Backup center > Start: Configure Backup >

Configure Backup ...

Basics **Backup policy** Datasources Review and configure

Choose a backup policy \*

DiskBackupPolicy

Create new

Selected policy details

Backup schedule Every 4 hours

Retention settings (in the order of priority) ⓘ

Retention rules	Operational data store	
First successful backup taken every day	30 Days	<a href="#">View details</a>
Default	7 Days	<a href="#">View details</a>

Previous Next

5. On the **Datasources** tab, click + Add/Edit to choose one or more Azure Managed Disks for which you want to configure backup.

The screenshot shows the 'Configure Backup' wizard in the Azure portal. The current step is 'Select resources to backup'. A table lists a single disk named 'DemoDataDisk' under the 'Subscription' column. Below the table, a note says 'Please ensure the pre-requisites are met before proceeding - Learn More'. Under 'Snapshot resource group', it shows a tree structure with 'Subscription' and 'Snapshot resource group' selected. A 'Validate' button is present. At the bottom, there are 'Previous' and 'Next' buttons, and a 'Select (0 items)' link.

#### NOTE

While the portal allows you to select multiple disks and configure backup, each disk is an individual backup instance. Currently, Azure Disk Backup only supports backup of individual disks. Point-in-time backup of multiple disks attached to a virtual machine isn't supported.

In the Azure portal, you can only select disks within the same subscription. If you have several disks to be backed up or if the disks reside in different subscriptions, you can use scripts ([PowerShell/CLI](#)) to automate.

See the [support matrix](#) for more information on the Azure Disk backup region availability, supported scenarios, and limitations.

## 6. Select Snapshot resource group and click Validate to initiate prerequisites checks.

Choosing resource group for storing and managing snapshots:

- Don't select the same resource group as that of the source disk.
- As a guideline, it's recommended to create a dedicated resource group as a snapshot datastore to be used by the Azure Backup service. Having a dedicated resource group allows restricting access permissions on the resource group, providing safety and ease of management of the backup data.
- You can use this resource group for storing snapshots across multiple disks that are being (or planned to be) backed up.
- You can't create an incremental snapshot for a particular disk outside of that disk's subscription. So, choose the resource group within the same subscription where the disk needs to be backed up. [Learn more](#) about incremental snapshot for managed disks.
- Once you configure the backup of a disk, you can't change the Snapshot Resource Group that's assigned to a backup instance.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

preview

Basics Backup policy Datasources Review and configure

Please ensure the pre-requisites are met before proceeding - [Learn more](#)

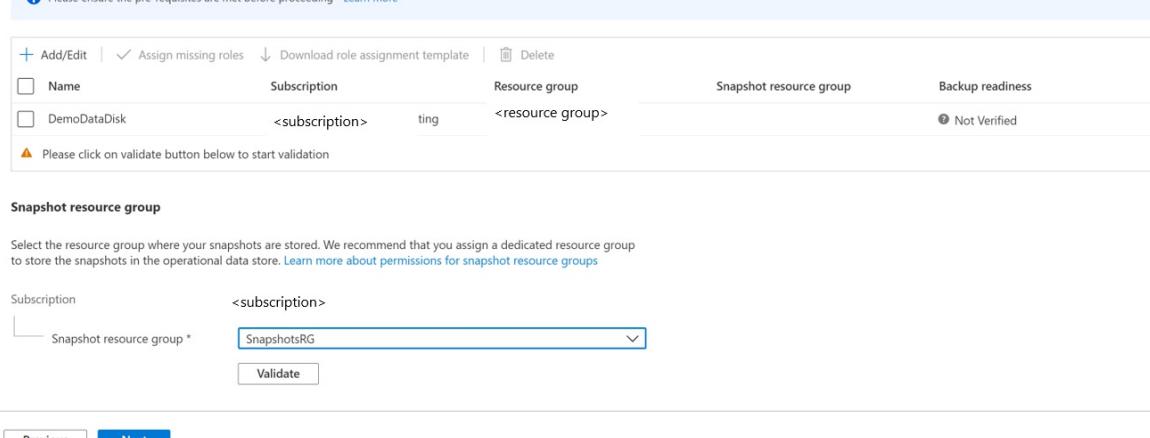
Add/Edit	Assign missing roles	Download role assignment template	Delete	
<input type="checkbox"/> Name	Subscription	Resource group	Snapshot resource group	Backup readiness
<input type="checkbox"/> DemoDataDisk	<subscription>	ting	<resource group>	<input checked="" type="radio"/> Not Verified

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription <subscription>  
Snapshot resource group \* SnapshotsRG

Previous Next



7. Once the validation is complete, check if there are any errors reported in the Backup readiness column.

### NOTE

Validation might take few minutes to complete. Validation may fail if:

- A disk is unsupported. See the [support matrix](#) for unsupported scenarios.
- The Backup vault managed identity does not have valid role assignments on the *disk* to be backed up or on the *snapshot resource group* where incremental snapshots are stored.

If the *Role assignment not done* error message displays in the **Backup readiness** column, the Backup vault managed identity needs role permissions on the selected disk(s) and/or on the Snapshot resource group.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

preview

Basics Backup policy Datasources Review and configure

Please ensure the pre-requisites are met before proceeding - [Learn more](#)

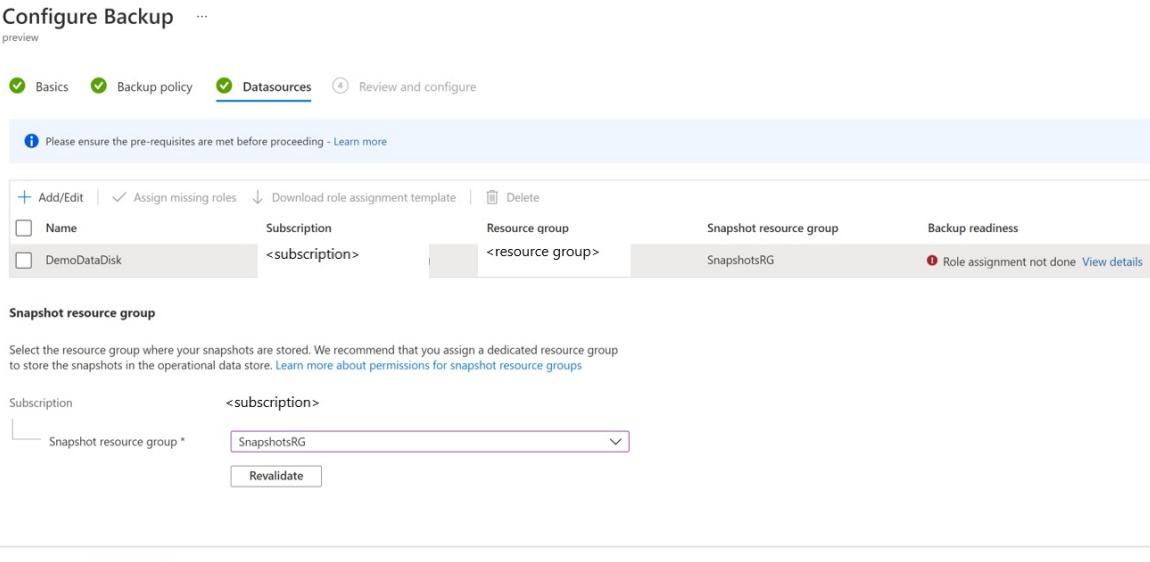
Add/Edit	Assign missing roles	Download role assignment template	Delete	
<input type="checkbox"/> Name	Subscription	Resource group	Snapshot resource group	Backup readiness
<input type="checkbox"/> DemoDataDisk	<subscription>	<resource group>	SnapshotsRG	<span style="color: red;">! Role assignment not done</span> <a href="#">View details</a>

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription <subscription>  
Snapshot resource group \* SnapshotsRG

Previous Next



To configure backup of managed disks, the following prerequisites are required:

#### NOTE

Backup vault uses managed identity to access other Azure resources. To configure a backup of managed disks, Backup Vault's managed identity requires a set of permissions on the source disks and resource groups where snapshots are created and managed.

A system-assigned managed identity is restricted to one per resource and is tied to the lifecycle of this resource. To grant permissions to the managed identity, use Azure role-based access control (Azure RBAC). Managed identity is a service principal of a special type that may only be used with Azure resources. Learn more about [managed identities](#).

- Assign the **Disk Backup Reader** role to Backup Vault's managed identity on the Source disk that needs to be backed up.
- Assign the **Disk Snapshot Contributor** role to the Backup vault's managed identity on the Resource group where backups are created and managed by the Azure Backup service. The disk snapshots are stored in a resource group within your subscription. To allow Azure Backup service to create, store, and manage snapshots, you need to provide permissions to the backup vault.

#### NOTE

The Configure Backup flow using Azure portal helps you in granting required role permissions to the above resources.

8. Select the checkbox next to each row with the *Role assignment not done* error message status in the Backup readiness column and click **Add missing roles** to automatically grant required role permissions for the Backup vault managed identity on selected resources.

Dashboard > Backup center > Start: Configure Backup >

Configure Backup ...

preview

Basics    Backup policy    Datasources    Review and configure

Please ensure the pre-requisites are met before proceeding - [Learn more](#)

Name	Subscription	Resource group	Snapshot resource group	Backup readiness
DemoDataDisk	<subscription>	<resource group>	SnapshotsRG	<span>Role assignment not done</span> <a href="#">View details</a>

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription: <subscription>

Snapshot resource group \*: SnapshotsRG

Revalidate

Previous    Next

9. Click **Confirm** to provide consent. Azure Backup will automatically propagate role assignment changes on your behalf and try to revalidate.

If you want to grant permission for the Backup vault managed identity to the selected disk(s) and snapshot resource group, select **Resource** in the **Scope** drop-down list.

Microsoft Azure (Preview) Search resources, services, and docs (G+ /)

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

Grant missing permissions

We will attempt to automatically propagate role assignment changes and try to revalidate. In some cases it can take up to 30 mins for the role assignment to propagate, resulting in revalidation failures. In such cases, please wait a few minutes before revalidating again.

Scope

DemoDataDisk <subscription> <resource group> SnapshotsRG ● Role assignment not done [View details](#)

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription <subscription>  
Snapshot resource group \*

### TIP

If you plan to configure backup for other disks in the same resource group/subscription in future, you can choose to provide permission at the scope of resource group or subscription.

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

... preview

Basics  Backup policy  **DataSources**  Review and configure

● Please ensure the pre-requisites are met before proceeding - [Learn more](#)

Name Subscription Resource group Snapshot resource group Backup readiness  
DemoDataDisk <subscription> <resource group> SnapshotsRG ● Role assignment not done [View details](#)

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription <subscription>  
Snapshot resource group \*

... Deployment in progress...  
Deployment to resource group 'ContosoRG' is in progress.

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

... preview

Basics  Backup policy  **DataSources**  Review and configure

● Please ensure the pre-requisites are met before proceeding - [Learn more](#)

Name Subscription Resource group Snapshot resource group Backup readiness  
DemoDataDisk <subscription> <resource group> SnapshotsRG ● Waiting for 39 seconds to revalidate [View details](#)

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription <subscription>  
Snapshot resource group \*

✓ Waiting for 39 seconds to revalidate  
Waiting for permissions to propagate before triggering auto-validation

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

preview

Basics    Backup policy    **Datasources**    Review and configure

Please ensure the pre-requisites are met before proceeding - [Learn more](#)

Add/Edit		Assign missing roles	Download role assignment template	Delete	Snapshot resource group	Backup readiness
<input checked="" type="checkbox"/>	Name	Subscription	Resource group	<resource group>	SnapshotsRG	 Validating..
<input checked="" type="checkbox"/>	DemoDataDisk	<subscription>				

 Please wait for validations to complete

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription: <subscription>  
Snapshot resource group \*: SnapshotsRG  
Validate

Previous    Next

**Revalidating**  
Validation is in progress. We will be doing 3 retries. If the validation still fails for any of these resources, please wait for a few minutes and retry validation.

### NOTE

- In some cases, it can take up to 30 minutes for the role assignments to propagate, causing revalidation failure. In this scenario, retry after some time.
- If the **Add missing roles** action fails to assign permissions with the error 'Insufficient permission for role assignment' in Backup readiness column, it indicates that you don't have the privilege to assign role permissions. Choose Download role assignment template to download role assignments as scripts and seek support from your IT Administrator to run the scripts to complete the prerequisites.

Microsoft Azure (Preview)    Search resources, services, and docs (G+)

Dashboard > Backup center > Start: Configure Backup >

## Configure Backup

preview

Basics    Backup policy    **Datasources**    Review and configure

Please ensure the pre-requisites are met before proceeding - [Learn more](#)

Add/Edit		Assign missing roles	Download role assignment template	Delete	Snapshot resource group	Backup readiness
<input checked="" type="checkbox"/>	Name	Subscription	Resource group	<resource group>	SnapshotsRG	 Success
<input checked="" type="checkbox"/>	DemoDataDisk	<subscription>				

 Validations have succeeded

**Snapshot resource group**

Select the resource group where your snapshots are stored. We recommend that you assign a dedicated resource group to store the snapshots in the operational data store. [Learn more about permissions for snapshot resource groups](#)

Subscription: <subscription>  
Snapshot resource group \*: SnapshotsRG  
Validate

Previous    Next

10. After a successful validation, click **Next** to move to the **Review and configure** tab, and then click **Configure backup** to configure backup of selected disks.

**Configure Backup** ...  
preview

Basics    Backup policy    Datasources    Review and configure

**Basics**

Datasource type	Azure Disks
Subscription	<subscription>
Location	southeastasia
Vault	DemoVault

**Policy**

Policy	DiskBackupPolicy
--------	------------------

**Resources**

1 resources			
Name	Subscription	Resource group	Snapshot resource ...
DemoDataDisk	<subscription>	<resource group>	SnapshotsRG

Previous    **Configure backup**

## Run an on-demand backup

1. In the **DemoVault Backup vault** created in the previous step, go to **Backup instances** and select a backup instance.

**DemoVault | Backup instances**

Subscription == <subscription>    Resource group == contosorg    Datasource type == Azure Disk    Vault == DemoVault    Protection

Name	Resource group	Backup policy	Protection status
ContosoDemoVM_DataDisk_0	ContosoRG	DiskBackupPolicy	Green

2. In the **Backup instances** screen, you'll find:

- **essential** information including source disk name, the snapshot resource group where incremental snapshots are stored, backup vault, and backup policy.
- **Job status** showing summary of backup and restore operations and their status in the last seven days.
- A list of **restore points** for the selected time period.

3. Select **Backup** to initiate an on-demand backup.

Microsoft Azure (Preview) Search resources, services, and docs (G+ /)

Home > DemoVault > ContosoDemoVM\_DataDisk\_0

Backup instance

Backup Now Restore Change policy Delete Refresh

**Essentials**

Datasource	: ContosoDemoVM_DataDisk_0	Location	: Southeast Asia
Datasource type	: Azure Disk	Status	: Protection configured
Snapshot resource group	: snapshotsrg	Backup Vault	: DemoVault
Subscription (change)	: <subscription>	Backup Policy	: DiskBackupPolicy
Subscription ID	: <subscription-id>	Backup storage redundan...	: Globally-redundant

See more

**Jobs (last 7 days)** View all

Operation	Failed	In progress	Completed
Scheduled backup	0	0	0
On-demand backup	0	0	0
Restore	0	0	0

**RESTORE POINTS**

Time period: Last 30 days

4. Select one of the retention rules associated with the backup policy. This retention rule will determine the retention duration of this on-demand backup. Select **Backup now** to start the backup.

Microsoft Azure (Preview) Search resources, services, and docs (G+ /)

Home > DemoVault > ContosoDemoVM\_DataDisk\_0 > Backup Now

ContosoDemoVM\_DataDisk\_0

The retention settings below are as per [DiskBackupPolicy](#) associated with the ContosoDemoVM\_DataDisk\_0 backup instance.

Select Retention Setting

Retention rules	Snapshot data store	Delete by	
Daily	30 Days	2/3/2021	<a href="#">View details</a>
Default	7 Days	1/11/2021	<a href="#">View details</a>

Backup now

## Track a backup operation

The Azure Backup service creates a job for scheduled backups or if you trigger on-demand backup operation for tracking. To view the backup job status:

1. Go to the **Backup instance** screen. It shows the jobs dashboard with operation and status for the past seven days.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > DemoVault > ContosoDemoVM\_DataDisk\_0

Backup instance

Backup Now Restore Change policy Delete Refresh

**Essentials**

Datasource	: ContosoDemoVM_DataDisk_0	Location	: Southeast Asia
Datasource type	: Azure Disk	Status	: Protection configured
Snapshot resource group	: snapshotsrg	Backup Vault	: DemoVault
Subscription (change)	: <subscription>	Backup Policy	: DiskBackupPolicy
Subscription ID	: <subscription-id>	Backup storage redund...	: Globally-redundant

See more

**Jobs (last 7 days)** View all

Operation	Failed	In progress	Completed
Scheduled backup	0	0	0
On-demand backup	0	1	0
Restore	0	0	0

**RESTORE POINTS**

Time period : Last 30 days

2. To view the status of the backup operation, select **View all** to show ongoing and past jobs of this backup instance.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > DemoVault > ContosoDemoVM\_DataDisk\_0 > Backup jobs

Refresh Select columns

Subscription == <subscription> Resource group == contosorg Datasource type == Azure Disk Vault == DemoVault Time range : Last week Status == In Progress

Operation == On-demand backup

Filter by backup instance

1-1 of 1 items

Backup instance	Operation	Status	Start time	Duration
ContosoDemoVM_DataDisk_0	On-demand backup	In progress	1/4/2021, 11:25:48 AM	00:01:33

< Previous 1 Next >

3. Review the list of backup and restore jobs and their status. Select a job from the list of jobs to view job details.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > DemoVault > ContosoDemoVM\_DataDisk\_0 > Backup jobs > Scheduled Backup

ContosoDemoVM\_DataDisk\_0

Refresh

**Job Details**

Activity ID	5f7b2a1f-f1ab-4abe-aadf-e7dc482381a8-lbz
Backup instance	ContosoDemoVM_DataDisk_0
Operation	Scheduled Backup
Status	In Progress
Start time	1/4/2021, 11:25:48 AM
Duration	00:02:07
User triggered job	true
Policy name	DiskBackupPolicy
RetentionTag	Default

**Sub tasks**

Name	Status
Trigger Backup	In progress

## Next steps

- [Restore Azure Managed Disks](#)

# Restore Azure Managed Disks

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article explains how to restore [Azure Managed Disks](#) from a restore point created by Azure Backup.

Currently, the Original-Location Recovery (OLR) option of restoring by replacing existing the source disk from where the backups were taken isn't supported. You can restore from a recovery point to create a new disk either in the same resource group as that of the source disk from where the backups were taken or in any other resource group. This is known as Alternate-Location Recovery (ALR) and this helps to keep both the source disk and the restored (new) disk.

In this article, you'll learn how to:

- Restore to create a new disk
- Track the restore operation status

## Restore to create a new disk

Backup Vault uses Managed Identity to access other Azure resources. To restore from backup, Backup vault's managed identity requires a set of permissions on the resource group where the disk is to be restored.

Backup vault uses a system assigned managed identity, which is restricted to one per resource and is tied to the lifecycle of this resource. You can grant permissions to the managed identity by using Azure role-based access control (Azure RBAC). Managed identity is a service principal of a special type that may only be used with Azure resources. Learn more about [Managed Identities](#).

The following pre-requisites are required to perform a restore operation:

1. Assign the **Disk Restore Operator** role to the Backup Vault's managed identity on the Resource group where the disk will be restored by the Azure Backup service.

### NOTE

You can choose the same resource group as that of the source disk from where backups are taken or to any other resource group within the same or a different subscription.

- a. Go to the resource group where the disk is to be restored to. For example, the resource group is *TargetRG*.
- b. Go to **Access control (IAM)** and select **Add role assignments**
- c. On the right context pane, select **Disk Restore Operator** in the **Role** dropdown list. Select the backup vault's managed identity and **Save**.

### TIP

Type the backup vault's name to select the vault's managed identity.

The screenshot shows the 'Access control (IAM)' blade for a resource group named 'TargetRG'. The left sidebar includes sections like Overview, Activity log, Access control (IAM), Tags, Events, Settings, Deployments, Security, Policies, Properties, Locks, Cost Management, Cost analysis, Cost alerts (preview), Budgets, Advisor recommendations, and Monitoring. The main area has tabs for Check access, Role assignments, Roles, Deny assignments, and Classic administrators. Under 'Check access', there's a 'My access' section with a 'View my access' button, and a 'Check access' section with a 'Find' dropdown set to 'User, group, or service principal' and a search bar. On the right, there are three cards: 'Grant access to this resource', 'View access to this resource', and 'View deny assignments'. A modal window titled 'Add role assignment' is overlaid, showing the 'Role' dropdown set to 'Disk Restore Operator', the 'Assign access to' dropdown set to 'User, group, or service principal', and the 'Select' dropdown showing 'DemoVault'. The 'Selected members' list contains 'DemoVault'. At the bottom of the modal are 'Save' and 'Discard' buttons.

2. Verify that the backup vault's managed identity has the right set of role assignments on the resource group where the disk will be restored.

a. Go to **Backup vault** - > **Identity** and select **Azure role assignments**

The screenshot shows the 'Identity' blade for a backup vault named 'DemoVault'. The left sidebar includes sections like Overview, Activity log, Tags, Manage (Properties, Identity, Backup policies, Backup instances, Locks), Monitoring + reporting (Alerts, Backup jobs), Automation, and Tasks (preview). The main area shows a 'System assigned (preview)' section with a status of 'On', an object ID of '670ea13c-1dbc-45f0-a019-02b8dc01f7e3', and a 'Permissions' section with a 'Azure role assignments' button highlighted with a red box. A note at the bottom states: 'This resource is registered with Azure Active Directory. The managed identity can be configured to allow access to other resources. Be careful when making changes to the account as it can result in failures.' At the top, there are 'Save', 'Discard', 'Refresh', and 'Get feedback?' buttons.

b. Verify that the role, resource name, and resource type appear correctly.

The screenshot shows the 'Azure role assignments' blade for the 'DemoVault' backup vault. The left sidebar includes sections like Home, Azure role assignments, Add role assignment (Preview), Refresh, and a note about permissions. The main area lists role assignments with columns for Subscription, Role, Resource Name, Resource Type, and Assigned To. Two entries are shown: 'Disk Snapshot Contributor' assigned to 'SnapshotsRG' as a Resource Group, and 'Disk Restore Operator' assigned to 'TargetRG' as a Resource Group. Both entries are highlighted with a red box. At the bottom, there are 'Save' and 'Discard' buttons.

Subscription *	Role	Resource Name	Resource Type	Assigned To
AzureBackup_Functional_Testing	Disk Snapshot Contributor	SnapshotsRG	Resource Group	DemoVault
	Disk Restore Operator	TargetRG	Resource Group	DemoVault
	Disk Backup Reader	ContosoDemoVM_DataDisk_0	Disks	DemoVault

#### NOTE

While the role assignments are reflected correctly on the portal, it may take approximately 15 minutes for the permission to be applied on the backup vault's managed identity.

During scheduled backups or an on-demand backup operation, Azure Backup stores the disk incremental snapshots in the Snapshot Resource Group provided during configuring backup of the disk. Azure Backup uses these incremental snapshots during the restore operation. If the snapshots are deleted or moved from the Snapshot Resource Group or if the Backup vault role assignments are revoked on the Snapshot Resource Group, the restore operation will fail.

3. If the disk to be restored is encrypted with [customer-managed keys \(CMK\)](#) or using [double encryption using platform-managed keys and customer-managed keys](#), then assign the **Reader** role permission to the Backup Vault's managed identity on the **Disk Encryption Set** resource.

Once the prerequisites are met, follow these steps to perform the restore operation.

1. In the [Azure portal](#), go to **Backup center**. Select **Backup instances** under the **Manage** section. From the list of backup instances, select the disk backup instance for which you want to perform the restore operation.

Name	Datasource subscription	Datasource resource group	Datasource location	Vault
ContosoDemoVM_DataDisk_0	AzureBackup_Functional_Testing	CortosoRG	Southeast Asia	DemoVault

Alternately, you can perform this operation from the Backup vault you used to configure backup for the disk.

2. In the **Backup instance** screen, select the restore point that you want to use to perform the restore operation and select **Restore**.

The screenshot shows the Microsoft Azure Backup center interface. At the top, there are navigation links: Home > Backup center (Preview) > ContosoDemoVM\_DataDisk\_0. Below this, a 'Backup instance' header includes options: Backup Now, Restore, Change policy, Delete, and Refresh. A 'Jobs (last 7 days)' table provides a summary of recent operations:

Operation	Failed	In progress	Completed
Scheduled backup	1	0	4
On-demand backup	2	0	0
Restore	0	0	0

Below the jobs table is a 'RESTORE POINTS' section. It shows a table of restore points from the last 30 days:

Time	Backup type	Data store	Action
1/5/2021, 5:02:15 AM	Incremental	Snapshot	Restore
1/5/2021, 1:02:21 AM	Incremental	Snapshot	...
1/4/2021, 9:01:58 PM	Incremental	Snapshot	...
1/4/2021, 5:12:31 PM	Incremental	Snapshot	...

3. In the **Restore** workflow, review the **Basics** and **Select recovery point** tab information, and select **Next: Restore parameters**.

The screenshot shows the 'Select recovery point' step of the restore workflow. The navigation bar at the top indicates the current step: ② Select recovery point. The main area displays the selected recovery point details:

Recovery Point \*: 1/5/2021, 5:02:15 AM  
Select recovery point

Data store: Snapshot

At the bottom, there are 'Previous' and 'Next: Restore parameters >' buttons.

4. In the **Restore parameters** tab, select the **Target subscription** and **Target resource group** where you want to restore the backup to. Provide the name of the disk to be restored. Select **Next: Review + restore**.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Backup center (Preview) > ContosoDemoVM\_DataDisk\_0 >

## Restore

Basics Select recovery point Restore parameters Review + restore

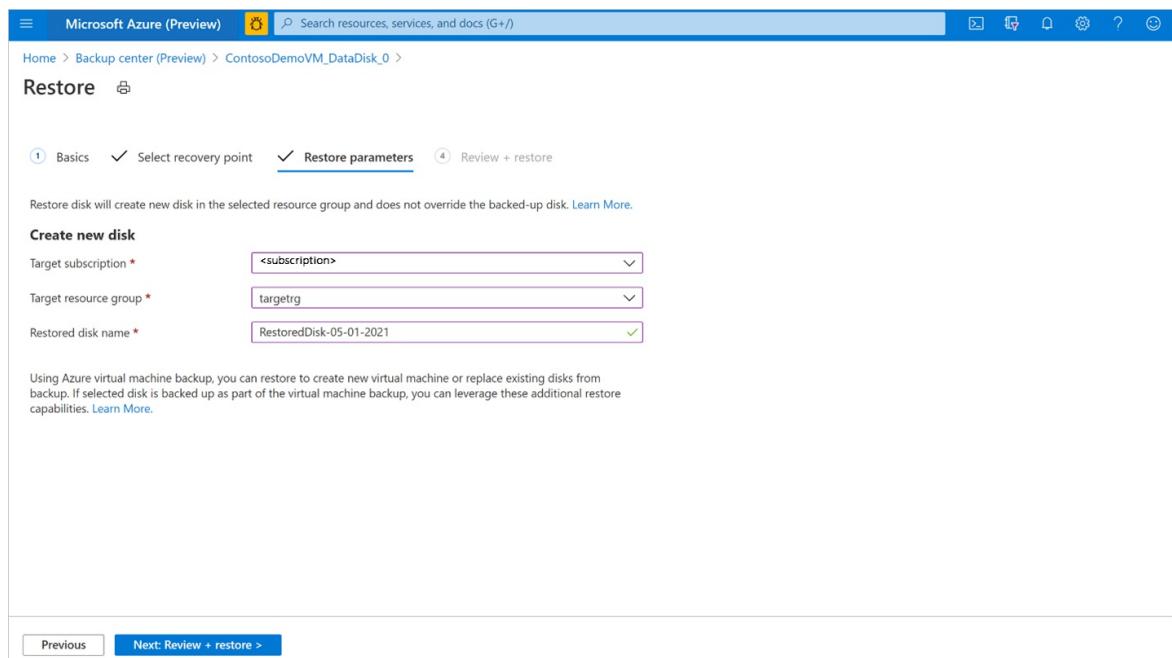
Restore disk will create new disk in the selected resource group and does not override the backed-up disk. [Learn More](#).

Create new disk

Target subscription \* <subscription>  
Target resource group \* targetrg  
Restored disk name \* RestoredDisk-05-01-2021

Using Azure virtual machine backup, you can restore to create new virtual machine or replace existing disks from backup. If selected disk is backed up as part of the virtual machine backup, you can leverage these additional restore capabilities. [Learn More](#).

Previous Next: Review + restore >

**TIP**

Disks being backed up by Azure Backup using the Disk Backup solution can also be backed up by Azure Backup using the Azure VM backup solution with the Recovery Services vault. If you have configured protection of the Azure VM to which this disk is attached, you can also use the Azure VM restore operation. You can choose to restore the VM, or disks and files or folders from the recovery point of the corresponding Azure VM backup instance. For more information, see [Azure VM backup](#).

- Once the validation is successful, select **Restore** to start the restore operation.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Backup center (Preview) > ContosoDemoVM\_DataDisk\_0 >

## Restore

Validation succeeded.

Basics Select recovery point Restore parameters Review + restore

Basics

Data source type Azure Disk  
Backup instance ContosoDemoVM\_DataDisk\_0

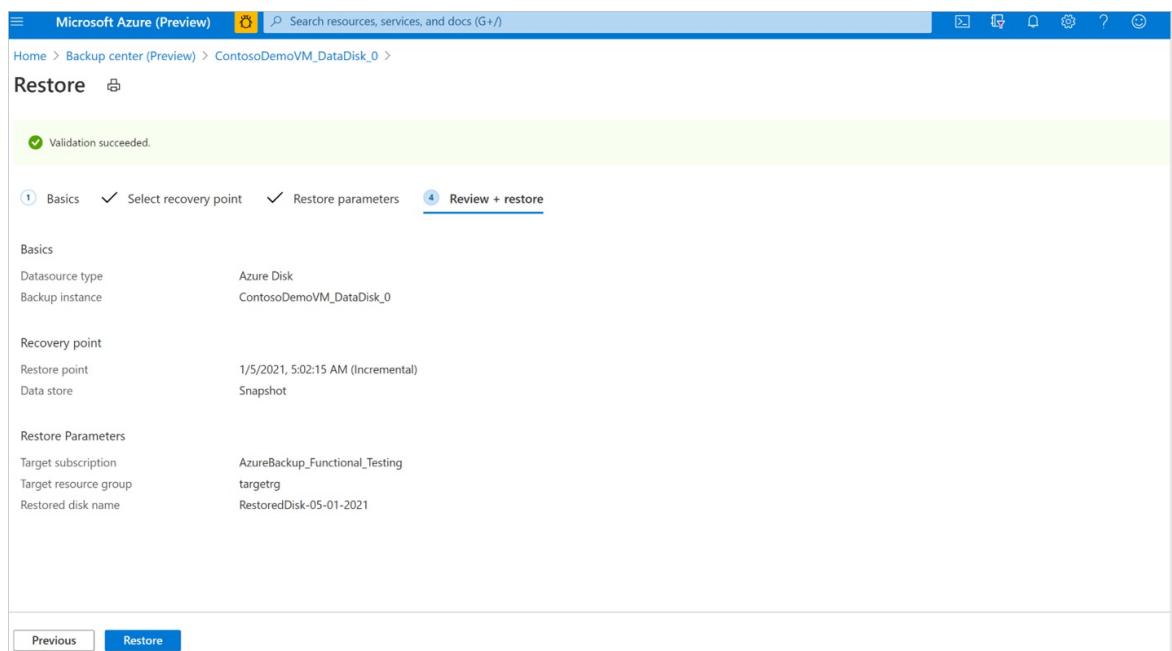
Recovery point

Restore point 1/5/2021, 5:02:15 AM (Incremental)  
Data store Snapshot

Restore Parameters

Target subscription AzureBackup\_Functional\_Testing  
Target resource group targetrg  
Restored disk name RestoredDisk-05-01-2021

Previous Restore



## NOTE

Validation might take few minutes to complete before you can trigger restore operation. Validation may fail if:

- a disk with the same name provided in **Restored disk name** already exists in the **Target resource group**
- the Backup vault's managed identity doesn't have valid role assignments on the **Target resource group**
- the Backup vault's managed identity role assignments are revoked on the **Snapshot resource group** where incremental snapshots are stored
- If incremental snapshots are deleted or moved from the snapshot resource group

Restore will create a new disk from the selected recovery point in the target resource group that was provided during the restore operation. To use the restored disk on an existing virtual machine, you'll need to perform more steps:

- If the restored disk is a data disk, you can attach an existing disk to a virtual machine. If the restored disk is OS disk, you can swap the OS disk of a virtual machine from the Azure portal under the **Virtual machine** pane - > **Disk**s menu in the **Settings** section.

Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (...)	Encryption
ContosoDemoVM_OsDisk_1_84b542ec3	Premium SSD	127	500	100	SSE with PMK

LUN	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (...)	Encryption
0	ContosoDemoVM_DataDisk_0	Premium SSD	1024	5000	200	SSE with PMK

- For Windows virtual machines, if the restored disk is a data disk, follow the instructions to [detach the original data disk](#) from the virtual machine. Then [attach the restored disk](#) to the virtual machine. Follow the instructions to [swap the OS disk](#) of the virtual machine with the restored disk.
- For Linux virtual machines, if the restored disk is a data disk, follow the instructions to [detach the original data disk](#) from the virtual machine. Then [attach the restored disk](#) to the virtual machine. Follow the instructions to [swap the OS disk](#) of the virtual machine with the restored disk.

It's recommended that you revoke the **Disk Restore Operator** role assignment from the Backup vault's managed identity on the **Target resource group** after the successful completion of restore operation.

## Track a restore operation

After you trigger the restore operation, the backup service creates a job for tracking. Azure Backup displays notifications about the job in the portal. To view the restore job progress:

1. Go to the **Backup instance** screen. It shows the jobs dashboard with operation and status for the past seven days.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Backup center (Preview) > ContosoDemoVM\_DataDisk\_0

Backup instance

Backup Now Restore Change policy Delete Refresh

Essentials

**Jobs (last 7 days)**

Operation	Failed	In progress	Completed
Scheduled backup	1	0	4
On-demand backup	2	0	0
Restore	0	1	0

**RESTORE POINTS**

Time period : Last 30 days

Time	Backup type	Data store
1/5/2021, 5:02:15 AM	Incremental	Snapshot
1/5/2021, 1:02:21 AM	Incremental	Snapshot
1/4/2021, 9:01:58 PM	Incremental	Snapshot
1/4/2021, 5:12:31 PM	Incremental	Snapshot

- To view the status of the restore operation, select **View all** to show ongoing and past jobs of this backup instance.

Microsoft Azure (Preview) Search resources, services, and docs (G+)

Home > Backup center (Preview) > DemoVault > ContosoDemoVM\_DataDisk\_0 >

Backup jobs

Refresh Select columns

Subscription == <subscription> Resource group == contosorg Datasource type == All Vault == DemoVault Time range : Last week Status == All Operation

Filter by backup instance

1-9 of 9 items

Backup instance	Operation	Status	Start time	Duration
ContosoDemoVM_DataDisk_0	Scheduled Backup	Completed	1/5/2021, 9:00:06 AM	00:03:03
ContosoDemoVM_DataDisk_0	Restore	Completed	1/5/2021, 7:33:32 AM	00:04:33
ContosoDemoVM_DataDisk_0	Scheduled Backup	Completed	1/5/2021, 5:00:03 AM	00:03:03
ContosoDemoVM_DataDisk_0	Scheduled Backup	Completed	1/5/2021, 1:00:00 AM	00:03:03
ContosoDemoVM_DataDisk_0	Scheduled Backup	Completed	1/4/2021, 9:00:08 PM	00:02:33
ContosoDemoVM_DataDisk_0	Scheduled Backup	Completed	1/4/2021, 5:00:04 PM	00:13:04
ContosoDemoVM_DataDisk_0	Scheduled Backup	Failed	1/4/2021, 1:00:01 PM	00:01:01
ContosoDemoVM_DataDisk_0	On-demand backup	Failed	1/4/2021, 12:46:02 PM	00:37:15
ContosoDemoVM_DataDisk_0	On-demand backup	Failed	1/4/2021, 11:25:48 AM	00:34:13

< Previous 1 Next >

- Review the list of backup and restore jobs and their status. Select a job from the list of jobs to view job details.

Microsoft Azure (Preview) Search resources, services, and docs (G+ /)

Home > Backup center (Preview) > DemoVault > ContosoDemoVM\_DataDisk\_0 > Backup jobs >

**Restore**

ContosoDemoVM\_DataDisk\_0

Refresh

Job Details

Activity ID	eb70a9ba-40e2-4fbf-8871-d065e99e6055-lbz
Backup instance	ContosoDemoVM_DataDisk_0
Operation	Restore
Status	Completed
Start time	1/5/2021, 7:33:32 AM
Duration	0:04:33
User triggered job	true
Policy name	DiskBackupPolicy
Recovery point time	1/5/2021, 5:02:15 AM
Recovery destination	/subscriptions/62b829ee-7936-40c9-a1c9-47a93f9f3965/resourceGroups/targetrg/providers/Microsoft.Compute/disks/RestoredDisk-05-01-2021

Sub tasks

Name	Status
Trigger Restore	Completed

## Next steps

- [Azure Disk Backup FAQ](#)

# Backup and disaster recovery for Azure managed disks

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article explains how to plan for backup and disaster recovery for Azure managed disks.

First, we cover the built-in fault tolerance capabilities in the Azure platform that guard against local failures. We then discuss the disaster scenarios not fully covered by the built-in capabilities. We also show several examples of workload scenarios where different backup and disaster recovery considerations can apply. We also cover some disaster recovery solutions for managed disks.

## Introduction

Azure uses various methods for redundancy and fault tolerance to protect customers from localized hardware failures. Local failures can include problems with an Azure Storage server machine that stores part of the data for a virtual disk or failures of solid-state drives (SSDs) or hard disk drives (HDDs) on that server. Isolated hardware component failures can happen during normal operations.

Azure is designed to be resilient to these failures. Major disasters can result in failures or the inaccessibility of many storage servers or even a whole data center. Although your virtual machines (VMs) and disks are normally protected from localized failures, additional steps are necessary to protect your workload from region-wide catastrophic failures, such as a major disaster, that can affect your VMs and disks.

In addition to the possibility of platform failures, problems with a customer application or data can occur. For example, a new version of your application might inadvertently make a change to the data that causes it to break. In that case, you might want to revert the application and the data to a prior version that contains the last known good state. This requires maintaining regular backups.

For regional disaster recovery, you must back up your infrastructure as a service (IaaS) VM disks to a different region.

Before we look at backup and disaster recovery options, let's recap a few methods available for handling localized failures.

### Azure IaaS resiliency

Resiliency refers to the tolerance for normal failures that occur in hardware components. Resiliency is the ability to recover from failures and continue to function. It's not about avoiding failures, but responding to failures in a way that avoids downtime or data loss. The goal of resiliency is to return the application to a fully functioning state following a failure. Azure VMs and managed disks are designed to be resilient to common hardware faults. Let's look at how the Azure IaaS platform provides this resiliency.

A VM consists mainly of two parts: a compute server and the persistent disks. Both affect the fault tolerance of a VM.

If the Azure compute host server that houses your VM experiences a hardware failure, which is rare, Azure is designed to automatically restore the VM on another server. In this scenario, your computer reboots, and the VM comes back up after some time. Azure automatically detects such hardware failures and begins recovery to help ensure the customer VM is available as soon as possible.

Regarding your managed disk, the durability of data is critical for a persistent storage platform. Azure customers

have important business applications running on IaaS, and they depend on the persistence of the data. Azure designs protection for these IaaS disks, with three redundant copies of the data that is stored locally. These copies provide for high durability against local failures. If one of the hardware components that holds your disk fails, your VM is not affected, because there are two additional copies to support disk requests. It works fine, even if two different hardware components that support a disk fail at the same time (which is rare).

To ensure that you always maintain three replicas, Azure automatically creates a new copy of the data in the background if one of the three copies becomes unavailable. Therefore, it should not be necessary to use RAID with Azure disks for fault tolerance. A simple RAID 0 configuration should be sufficient for striping the disks, if necessary, to create larger volumes.

Because of this architecture, Azure has consistently delivered enterprise-grade durability for IaaS disks, with an industry-leading zero percent [annualized failure rate](#).

Localized hardware faults on the compute host or in the storage platform can sometimes result in the temporary unavailability of the VM that is covered by the [Azure SLA](#) for VM availability. Azure also provides an industry-leading SLA for single VM instances that use Azure premium SSDs.

To safeguard application workloads from downtime due to the temporary unavailability of a disk or VM, customers can use [availability sets](#). Two or more virtual machines in an availability set provide redundancy for the application. Azure then creates these VMs and disks in separate fault domains with different power, network, and server components.

Because of these separate fault domains, localized hardware failures typically don't affect multiple VMs in the set at the same time. Having separate fault domains provides high availability for your application. It's considered a good practice to use availability sets when high availability is required.

### **Backup and disaster recovery**

Disaster recovery is the ability to recover from rare, but major, incidents. These incidents include non-transient, wide-scale failures, such as a service disruption that affects an entire region. Disaster recovery includes data backup and archiving, and might include manual intervention, such as restoring a database from a backup.

The Azure platform's built-in protection against localized failures might not fully protect the VMs/disks if a major disaster causes large-scale outages. These large-scale outages include catastrophic events, such as if a data center is hit by a hurricane, earthquake, fire, or if there is a large-scale hardware unit failure. In addition, you might encounter failures due to application or data issues.

To help protect your IaaS workloads from outages, you should plan for redundancy and have backups to enable recovery. For disaster recovery, you should back up in a different geographic location away from the primary site. This approach helps ensure your backup is not affected by the same event that originally affected the VM or disks. For more information, see [Disaster recovery for Azure applications](#).

Your disaster recovery considerations might include the following aspects:

- **High availability:** The ability of the application to continue running in a healthy state, without significant downtime. By healthy state, this state means that the application is responsive, and users can connect to the application and interact with it. Certain mission-critical applications and databases might be required to always be available, even when there are failures in the platform. For these workloads, you might need to plan redundancy for the application, as well as the data.
- **Data durability:** In some cases, the main consideration is ensuring that the data is preserved if a disaster happens. Therefore, you might need a backup of your data in a different site. For such workloads, you might not need full redundancy for the application, but only a regular backup of the disks.

## **Backup and disaster recovery scenarios**

Let's look at a few typical examples of application workload scenarios and the considerations for planning for

disaster recovery.

### **Scenario 1: Major database solutions**

Consider a production database server, like SQL Server or Oracle, that can support high availability. Critical production applications and users depend on this database. The disaster recovery plan for this system might need to support the following requirements:

- The data must be protected and recoverable.
- The server must be available for use.

The disaster recovery plan might require maintaining a replica of the database in a different region as a backup. Depending on the requirements for server availability and data recovery, the solution might range from an active-active or active-passive replica site to periodic offline backups of the data. Relational databases, such as SQL Server and Oracle, provide various options for replication. For SQL Server, use [SQL Server Always On Availability Groups](#) for high availability.

NoSQL databases, like MongoDB, also support [replicas](#) for redundancy. The replicas for high availability are used.

### **Scenario 2: A cluster of redundant VMs**

Consider a workload handled by a cluster of VMs that provide redundancy and load balancing. One example is a Cassandra cluster deployed in a region. This type of architecture already provides a high level of redundancy within that region. However, to protect the workload from a regional-level failure, you should consider spreading the cluster across two regions or making periodic backups to another region.

### **Scenario 3: IaaS application workload**

Let's look at the IaaS application workload. For example, this application might be a typical production workload running on an Azure VM. It might be a web server or file server holding the content and other resources of a site. It might also be a custom-built business application running on a VM that stored its data, resources, and application state on the VM disks. In this case, it's important to make sure you take backups on a regular basis. Backup frequency should be based on the nature of the VM workload. For example, if the application runs every day and modifies data, then the backup should be taken every hour.

Another example is a reporting server that pulls data from other sources and generates aggregated reports. The loss of this VM or disks might lead to the loss of the reports. However, it might be possible to rerun the reporting process and regenerate the output. In that case, you don't really have a loss of data, even if the reporting server is hit with a disaster. As a result, you might have a higher level of tolerance for losing part of the data on the reporting server. In that case, less frequent backups are an option to reduce costs.

### **Scenario 4: IaaS application data issues**

IaaS application data issues are another possibility. Consider an application that computes, maintains, and serves critical commercial data, such as pricing information. A new version of your application had a software bug that incorrectly computed the pricing and corrupted the existing commerce data served by the platform. Here, the best course of action is to revert to the earlier version of the application and the data. To enable this, take periodic backups of your system.

## **Disaster recovery solution: Azure Disk Backup**

Azure Disk Backup is a native, cloud-based backup solution that protects your data in managed disks. It's a simple, secure, and cost-effective solution that enables you to configure protection for managed disks in a few steps. It assures that you can recover your data in a disaster scenario.

Azure Disk Backup offers a turnkey solution that provides snapshot lifecycle management for managed disks by automating periodic creation of snapshots and retaining it for configured duration using backup policy. You can manage the disk snapshots with zero infrastructure cost and without the need for custom scripting or any

management overhead. This is a crash-consistent backup solution that takes point-in-time backup of a managed disk using incremental snapshots with support for multiple backups per day. It's also an agent-less solution and doesn't impact production application performance. It supports backup and restore of both OS and data disks (including shared disks), whether or not they're currently attached to a running Azure VM.

For more details on Azure Disk Backup, see [Overview of Azure Disk Backup](#).

## Alternative solution: Consistent snapshots

If you are unable to use Azure Backup, you can implement your own backup mechanism by using snapshots. Creating consistent snapshots for all the disks used by a VM and then replicating those snapshots to another region is complicated. For this reason, Azure considers using the Backup service as a better option than building a custom solution.

If you use locally redundant storage for disks, you need to replicate the data yourself.

A snapshot is a representation of an object at a specific point in time. A snapshot incurs billing for the incremental size of the data it holds. For more information, see [Create an incremental snapshot for managed disks](#).

### Create snapshots while the VM is running

Although you can take a snapshot at any time, if the VM is running, there is still data being streamed to the disks. The snapshots might contain partial operations that were in flight. Also, if there are several disks involved, the snapshots of different disks might have occurred at different times. These scenarios may cause the snapshots to be uncoordinated. This lack of coordination is especially problematic for striped volumes whose files might be corrupted if changes were being made during backup.

To avoid this situation, the backup process must implement the following steps:

1. Freeze all the disks.
2. Flush all the pending writes.
3. [Create an incremental snapshot for managed disks](#) for all the disks.

Some Windows applications, like SQL Server, provide a coordinated backup mechanism via a volume shadow service to create application-consistent backups. On Linux, you can use a tool like `fsfreeze` for coordinating the disks. This tool provides file-consistent backups, but not application-consistent snapshots. This process is complex, so you should consider using [Azure Disk Backup](#) or a third-party backup solution that already implements this procedure.

The previous process results in a collection of coordinated snapshots for all of the VM disks, representing a specific point-in-time view of the VM. This is a backup restore point for the VM. You can repeat the process at scheduled intervals to create periodic backups. See [Copy the backups to another region](#) for steps to copy the snapshots to another region for disaster recovery.

### Create snapshots while the VM is offline

Another option to create consistent backups is to shut down the VM and take snapshots of each disk. Taking offline snapshots is easier than coordinating snapshots of a running VM, but it requires a few minutes of downtime.

### Copy the snapshots to another region

Creation of the snapshots alone might not be sufficient for disaster recovery. You must also copy the snapshots to another region. See [Copy an incremental snapshot to a new region](#).

## Other options

## **SQL Server**

SQL Server running in a VM has its own built-in capabilities to back up your SQL Server database to Azure Blob storage or a file share. For more information, see [Back up and restore for SQL Server in Azure virtual machines](#). In addition to back up and restore, [SQL Server Always On availability groups](#) can maintain secondary replicas of databases. This ability greatly reduces the disaster recovery time.

## Next steps

See [Back up Azure unmanaged Virtual Machine disks with incremental snapshots](#).

# Create a snapshot of a virtual hard disk

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

A snapshot is a full, read-only copy of a virtual hard disk (VHD). You can use a snapshot as a point-in-time backup, or to help troubleshoot virtual machine (VM) issues. You can take a snapshot of both operating system (OS) or data disk VHDs.

## Create a snapshot of a VHD

If you want to use a snapshot to create a new VM, ensure that you first cleanly shut down the VM. This action clears any processes that are in progress.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

To create a snapshot using the Azure portal, complete these steps.

1. In the [Azure portal](#), select **Create a resource**.
2. Search for and select **Snapshot**.
3. In the **Snapshot** window, select **Create**. The **Create snapshot** window appears.
4. For **Resource group**, select an existing [resource group](#) or enter the name of a new one.
5. Enter a **Name**, then select a **Region** and **Snapshot type** for the new snapshot. If you would like to store your snapshot in zone-resilient storage, you need to select a region that supports [availability zones](#). For a list of supporting regions, see [Azure regions with availability zones](#).
6. For **Source subscription**, select the subscription that contains the managed disk to be backed up.
7. For **Source disk**, select the managed disk to snapshot.
8. For **Storage type**, select **Standard HDD**, unless you require zone-redundant storage or high-performance storage for your snapshot.
9. If needed, configure settings on the **Encryption**, **Networking**, and **Tags** tabs. Otherwise, default settings are used for your snapshot.
10. Select **Review + create**.

## Next steps

To recover using a snapshot, you must create a new disk from the snapshot, then either deploy a new VM, and use the managed disk as the OS disk, or attach the disk as a data disk to an existing VM.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

For more information, see the example in [Create a VM from a VHD by using the Azure portal](#).

# Back up Azure unmanaged Virtual Machine disks with incremental snapshots

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

## Overview

Azure Storage provides the capability to take snapshots of blobs. Snapshots capture the blob state at that point in time. In this article, we describe a scenario in which you can maintain backups of virtual machine disks using snapshots. You can use this methodology when you choose not to use Azure Backup and Recovery Service, and wish to create a custom backup strategy for your virtual machine disks. For virtual machines running business or mission critical workloads, it's recommended to use [Azure Backup](#) as part of the backup strategy.

Azure virtual machine disks are stored as page blobs in Azure Storage. Since we are describing a backup strategy for virtual machine disks in this article, we refer to snapshots in the context of page blobs. To learn more about snapshots, refer to [Creating a Snapshot of a Blob](#).

## What is a snapshot?

A blob snapshot is a read-only version of a blob that is captured at a point in time. Once a snapshot has been created, it can be read, copied, or deleted, but not modified. Snapshots provide a way to back up a blob as it appears at a moment in time. Until REST version 2015-04-05, you had the ability to copy full snapshots. With the REST version 2015-07-08 and above, you can also copy incremental snapshots.

## Full snapshot copy

Snapshots can be copied to another storage account as a blob to keep backups of the base blob. You can also copy a snapshot over its base blob, which is like restoring the blob to an earlier version. When a snapshot is copied from one storage account to another, it occupies the same space as the base page blob. Therefore, copying whole snapshots from one storage account to another is slow and consumes much space in the target storage account.

### NOTE

If you copy the base blob to another destination, the snapshots of the blob are not copied along with it. Similarly, if you overwrite a base blob with a copy, snapshots associated with the base blob are not affected and stay intact under the base blob name.

## Back up disks using snapshots

As a backup strategy for your virtual machine disks, you can take periodic snapshots of the disk or page blob, and copy them to another storage account using tools like [Copy Blob](#) operation or [AzCopy](#). You can copy a snapshot to a destination page blob with a different name. The resulting destination page blob is a writeable page blob and not a snapshot. Later in this article, we describe steps to take backups of virtual machine disks using snapshots.

## Restore disks using snapshots

When it is time to restore your disk to a stable version that was previously captured in one of the backup snapshots, you can copy a snapshot over the base page blob. After the snapshot is promoted to the base page

blob, the snapshot remains, but its source is overwritten with a copy that can be both read and written. Later in this article we describe steps to restore a previous version of your disk from its snapshot.

## Implementing full snapshot copy

You can implement a full snapshot copy by doing the following,

- First, take a snapshot of the base blob using the [Snapshot Blob](#) operation.
- Then, copy the snapshot to a target storage account using [Copy Blob](#).
- Repeat this process to maintain backup copies of your base blob.

## Incremental snapshot copy

The new feature in the [GetPageRanges](#) API provides a much better way to back up the snapshots of your page blobs or disks. The API returns the list of changes between the base blob and the snapshots, which reduces the amount of storage space used on the backup account. The API supports page blobs on Premium Storage as well as Standard Storage. Using this API, you can build faster and more efficient backup solutions for Azure VMs. This API will be available with the REST version 2015-07-08 and higher.

Incremental Snapshot Copy allows you to copy from one storage account to another the difference between,

- Base blob and its Snapshot OR
- Any two snapshots of the base blob

Provided the following conditions are met,

- The blob was created on Jan-1-2016 or later.
- The blob was not overwritten with [PutPage](#) or [Copy Blob](#) between two snapshots.

### NOTE

This feature is available for Premium and Standard Azure Page Blobs.

When you have a custom backup strategy using snapshots, copying the snapshots from one storage account to another can be slow and can consume much storage space. Instead of copying the entire snapshot to a backup storage account, you can write the difference between consecutive snapshots to a backup page blob. This way, the time to copy and the space to store backups is substantially reduced.

## Implementing Incremental Snapshot Copy

You can implement incremental snapshot copy by doing the following,

- Take a snapshot of the base blob using [Snapshot Blob](#).
- Copy the snapshot to the target backup storage account in same or any other Azure region using [Copy Blob](#). This is the backup page blob. Take a snapshot of the backup page blob and store it in the backup account.
- Take another snapshot of the base blob using [Snapshot Blob](#).
- Get the difference between the first and second snapshots of the base blob using [GetPageRanges](#). Use the new parameter `prevsnapshot`, to specify the `DateTime` value of the snapshot you want to get the difference with. When this parameter is present, the REST response includes only the pages that were changed between target snapshot and previous snapshot including clear pages.
- Use [PutPage](#) to apply these changes to the backup page blob.
- Finally, take a snapshot of the backup page blob and store it in the backup storage account.

In the next section, we will describe in more detail how you can maintain backups of disks using Incremental Snapshot Copy

## Scenario

In this section, we describe a scenario that involves a custom backup strategy for virtual machine disks using snapshots.

Consider a DS-series Azure VM with a premium storage P30 disk attached. The P30 disk called *mypremiumdisk* is stored in a premium storage account called *mypremiumaccount*. A standard storage account called *mybackupsstdaccount* is used for storing the backup of *mypremiumdisk*. We would like to keep a snapshot of *mypremiumdisk* every 12 hours.

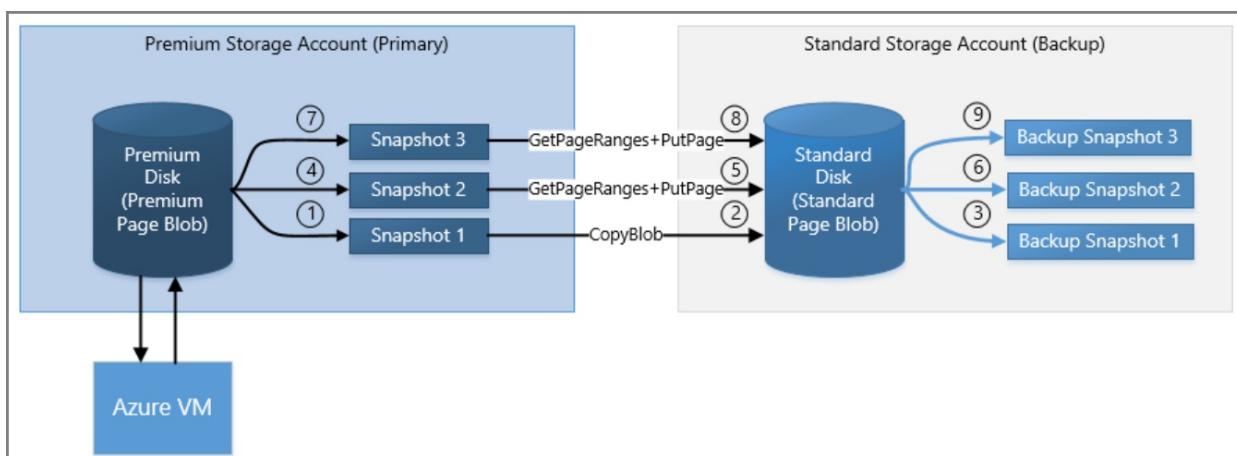
To learn about creating a storage account, see [Create a storage account](#).

To learn about backing up Azure VMs, refer to [Plan Azure VM backups](#).

## Steps to maintain backups of a disk using incremental snapshots

The following steps describe how to take snapshots of *mypremiumdisk* and maintain the backups in *mybackupsstdaccount*. The backup is a standard page blob called *mybackupsstdpageblob*. The backup page blob always reflects the same state as the last snapshot of *mypremiumdisk*.

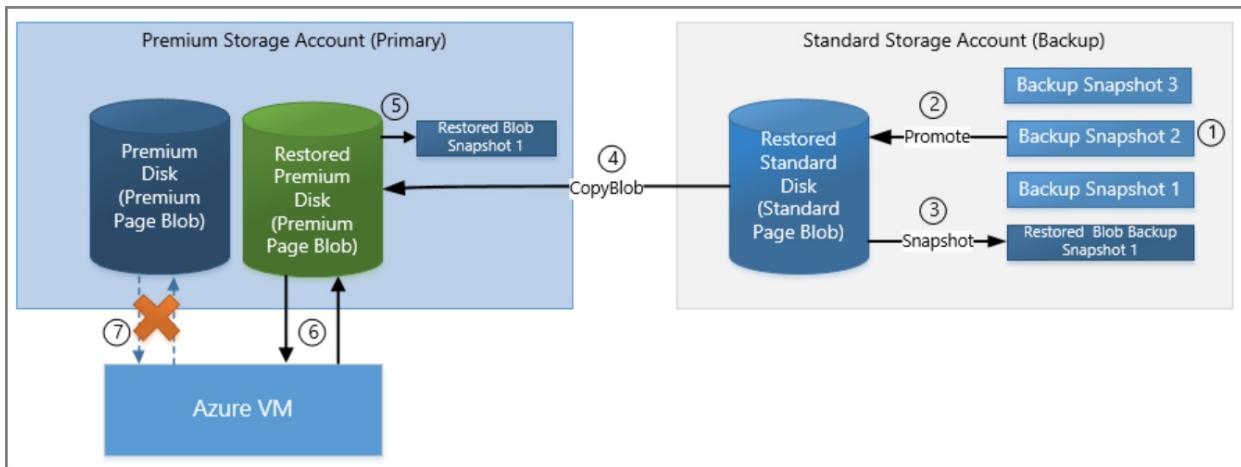
1. Create the backup page blob for your premium storage disk, by taking a snapshot of *mypremiumdisk* called *mypremiumdisk\_ss1*.
2. Copy this snapshot to *mybackupsstdaccount* as a page blob called *mybackupsstdpageblob*.
3. Take a snapshot of *mybackupsstdpageblob* called *mybackupsstdpageblob\_ss1*, using [Snapshot Blob](#) and store it in *mybackupsstdaccount*.
4. During the backup window, create another snapshot of *mypremiumdisk*, say *mypremiumdisk\_ss2*, and store it in *mypremiumaccount*.
5. Get the incremental changes between the two snapshots, *mypremiumdisk\_ss2* and *mypremiumdisk\_ss1*, using [GetPageRanges](#) on *mypremiumdisk\_ss2* with the **prevsnapshot** parameter set to the timestamp of *mypremiumdisk\_ss1*. Write these incremental changes to the backup page blob *mybackupsstdpageblob* in *mybackupsstdaccount*. If there are deleted ranges in the incremental changes, they must be cleared from the backup page blob. Use [PutPage](#) to write incremental changes to the backup page blob.
6. Take a snapshot of the backup page blob *mybackupsstdpageblob*, called *mybackupsstdpageblob\_ss2*. Delete the previous snapshot *mypremiumdisk\_ss1* from premium storage account.
7. Repeat steps 4-6 every backup window. In this way, you can maintain backups of *mypremiumdisk* in a standard storage account.



## Steps to restore a disk from snapshots

The following steps, describe how to restore the premium disk, *mypremiumdisk* to an earlier snapshot from the backup storage account *mybackupsstdaccount*.

1. Identify the point in time that you wish to restore the premium disk to. Let's say that it is snapshot *mybackupstdpageblob\_ss2*, which is stored in the backup storage account *mybackupstdaccount*.
2. In *mybackupstdaccount*, promote the snapshot *mybackupstdpageblob\_ss2* as the new backup base page blob *mybackupstdpageblobrestored*.
3. Take a snapshot of this restored backup page blob, called *mybackupstdpageblobrestored\_ss1*.
4. Copy the restored page blob *mybackupstdpageblobrestored* from *mybackupstdaccount* to *mypremiumaccount* as the new premium disk *mypremiumdiskrestored*.
5. Take a snapshot of *mypremiumdiskrestored*, called *mypremiumdiskrestored\_ss1* for making future incremental backups.
6. Point the DS series VM to the restored disk *mypremiumdiskrestored* and detach the old *mypremiumdisk* from the VM.
7. Begin the Backup process described in previous section for the restored disk *mypremiumdiskrestored*, using the *mybackupstdpageblobrestored* as the backup page blob.



## Next Steps

Use the following links to learn more about creating snapshots of a blob and planning your VM backup infrastructure.

- [Creating a Snapshot of a Blob](#)
- [Plan your VM Backup Infrastructure](#)

# Backup and disaster recovery for Azure unmanaged disks

9/21/2022 • 21 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article explains how to plan for backup and disaster recovery (DR) of IaaS virtual machines (VMs) and disks in Azure. This document covers unmanaged disks, or page blobs.

First, we cover the built-in fault tolerance capabilities in the Azure platform that helps guard against local failures. We then discuss the disaster scenarios not fully covered by the built-in capabilities. We also show several examples of workload scenarios where different backup and DR considerations can apply. We then review possible solutions for the DR of IaaS disks.

## Introduction

The Azure platform uses various methods for redundancy and fault tolerance to help protect customers from localized hardware failures. Local failures can include problems with an Azure Storage server machine that stores part of the data for a virtual disk or failures of an SSD or HDD on that server. Such isolated hardware component failures can happen during normal operations.

The Azure platform is designed to be resilient to these failures. Major disasters can result in failures or the inaccessibility of many storage servers or even a whole datacenter. Although your VMs and disks are normally protected from localized failures, additional steps are necessary to protect your workload from region-wide catastrophic failures, such as a major disaster, that can affect your VM and disks.

In addition to the possibility of platform failures, problems with a customer application or data can occur. For example, a new version of your application might inadvertently make a change to the data that causes it to break. In that case, you might want to revert the application and the data to a prior version that contains the last known good state. This requires maintaining regular backups.

For regional disaster recovery, you must back up your IaaS VM disks to a different region.

Before we look at backup and DR options, let's recap a few methods available for handling localized failures.

### Azure IaaS resiliency

*Resiliency* refers to the tolerance for normal failures that occur in hardware components. Resiliency is the ability to recover from failures and continue to function. It's not about avoiding failures, but responding to failures in a way that avoids downtime or data loss. The goal of resiliency is to return the application to a fully functioning state following a failure. Azure virtual machines and disks are designed to be resilient to common hardware faults. Let's look at how the Azure IaaS platform provides this resiliency.

A virtual machine consists mainly of two parts: a compute server and the persistent disks. Both affect the fault tolerance of a virtual machine.

If the Azure compute host server that houses your VM experiences a hardware failure, which is rare, Azure is designed to automatically restore the VM on another server. If this scenario, your computer reboots, and the VM comes back up after some time. Azure automatically detects such hardware failures and executes recoveries to help ensure the customer VM is available as soon as possible.

Regarding IaaS disks, the durability of data is critical for a persistent storage platform. Azure customers have important business applications running on IaaS, and they depend on the persistence of the data. Azure designs

protection for these IaaS disks, with three redundant copies of the data that is stored locally. These copies provide for high durability against local failures. If one of the hardware components that holds your disk fails, your VM is not affected, because there are two additional copies to support disk requests. It works fine, even if two different hardware components that support a disk fail at the same time (which is rare).

To ensure that you always maintain three replicas, Azure Storage automatically spawns a new copy of the data in the background if one of the three copies becomes unavailable. Therefore, it should not be necessary to use RAID with Azure disks for fault tolerance. A simple RAID 0 configuration should be sufficient for striping the disks, if necessary, to create larger volumes.

Because of this architecture, Azure has consistently delivered enterprise-grade durability for IaaS disks, with an industry-leading zero percent [annualized failure rate](#).

Localized hardware faults on the compute host or in the Storage platform can sometimes result in the temporary unavailability of the VM that is covered by the [Azure SLA](#) for VM availability. Azure also provides an industry-leading SLA for single VM instances that use Azure premium SSDs.

To safeguard application workloads from downtime due to the temporary unavailability of a disk or VM, customers can use [availability sets](#). Two or more virtual machines in an availability set provide redundancy for the application. Azure then creates these VMs and disks in separate fault domains with different power, network, and server components.

Because of these separate fault domains, localized hardware failures typically do not affect multiple VMs in the set at the same time. Having separate fault domains provides high availability for your application. It's considered a good practice to use availability sets when high availability is required. The next section covers the disaster recovery aspect.

### **Backup and disaster recovery**

Disaster recovery is the ability to recover from rare, but major, incidents. These incidents include non-transient, wide-scale failures, such as service disruption that affects an entire region. Disaster recovery includes data backup and archiving, and might include manual intervention, such as restoring a database from a backup.

The Azure platform's built-in protection against localized failures might not fully protect the VMs/disks if a major disaster causes large-scale outages. These large-scale outages include catastrophic events, such as if a datacenter is hit by a hurricane, earthquake, fire, or if there is a large-scale hardware unit failure. In addition, you might encounter failures due to application or data issues.

To help protect your IaaS workloads from outages, you should plan for redundancy and have backups to enable recovery. For disaster recovery, you should back up in a different geographic location away from the primary site. This approach helps ensure your backup is not affected by the same event that originally affected the VM or disks. For more information, see [Disaster recovery for Azure applications](#).

Your DR considerations might include the following aspects:

- High availability: The ability of the application to continue running in a healthy state, without significant downtime. By *healthy state*, this state means that the application is responsive, and users can connect to the application and interact with it. Certain mission-critical applications and databases might be required to always be available, even when there are failures in the platform. For these workloads, you might need to plan redundancy for the application, as well as the data.
- Data durability: In some cases, the main consideration is ensuring that the data is preserved if a disaster happens. Therefore, you might need a backup of your data in a different site. For such workloads, you might not need full redundancy for the application, but only a regular backup of the disks.

## **Backup and DR scenarios**

Let's look at a few typical examples of application workload scenarios and the considerations for planning for

disaster recovery.

### Scenario 1: Major database solutions

Consider a production database server, like SQL Server or Oracle, that can support high availability. Critical production applications and users depend on this database. The disaster recovery plan for this system might need to support the following requirements:

- The data must be protected and recoverable.
- The server must be available for use.

The disaster recovery plan might require maintaining a replica of the database in a different region as a backup. Depending on the requirements for server availability and data recovery, the solution might range from an active-active or active-passive replica site to periodic offline backups of the data. Relational databases, such as SQL Server and Oracle, provide various options for replication. For SQL Server, use [SQL Server Always On Availability Groups](#) for high availability.

NoSQL databases, like MongoDB, also support [replicas](#) for redundancy. The replicas for high availability are used.

### Scenario 2: A cluster of redundant VMs

Consider a workload handled by a cluster of VMs that provide redundancy and load balancing. One example is a Cassandra cluster deployed in a region. This type of architecture already provides a high level of redundancy within that region. However, to protect the workload from a regional-level failure, you should consider spreading the cluster across two regions or making periodic backups to another region.

### Scenario 3: IaaS application workload

Let's look at the IaaS application workload. For example, this application might be a typical production workload running on an Azure VM. It might be a web server or file server holding the content and other resources of a site. It might also be a custom-built business application running on a VM that stored its data, resources, and application state on the VM disks. In this case, it's important to make sure you take backups on a regular basis. Backup frequency should be based on the nature of the VM workload. For example, if the application runs every day and modifies data, then the backup should be taken every hour.

Another example is a reporting server that pulls data from other sources and generates aggregated reports. The loss of this VM or disks might lead to the loss of the reports. However, it might be possible to rerun the reporting process and regenerate the output. In that case, you don't really have a loss of data, even if the reporting server is hit with a disaster. As a result, you might have a higher level of tolerance for losing part of the data on the reporting server. In that case, less frequent backups are an option to reduce costs.

### Scenario 4: IaaS application data issues

IaaS application data issues are another possibility. Consider an application that computes, maintains, and serves critical commercial data, such as pricing information. A new version of your application had a software bug that incorrectly computed the pricing and corrupted the existing commerce data served by the platform. Here, the best course of action is to revert to the earlier version of the application and the data. To enable this, take periodic backups of your system.

## Disaster recovery solution: Azure Backup

[Azure Backup](#) is used for backups and DR, and it works with [managed disks](#) as well as unmanaged disks. You can create a backup job with time-based backups, easy VM restoration, and backup retention policies.

If you use [premium SSDs](#), [managed disks](#), or other disk types with the [locally redundant storage](#) option, it's especially important to make periodic DR backups. Azure Backup stores the data in your recovery services vault for long-term retention. Choose the [geo-redundant storage](#) option for the backup recovery services vault. That option ensures that backups are replicated to a different Azure region for safeguarding from regional disasters.

For unmanaged disks, you can use the locally redundant storage type for IaaS disks, but ensure that Azure Backup is enabled with the geo-redundant storage option for the recovery services vault.

**NOTE**

If you use the [geo-redundant storage](#) or [read-access geo-redundant storage](#) option for your unmanaged disks, you still need consistent snapshots for backup and DR. Use either [Azure Backup](#) or [consistent snapshots](#).

The following table is a summary of the solutions available for DR.

SCENARIO	AUTOMATIC REPLICATION	DR SOLUTION
Premium SSD disks	Local ( <a href="#">locally redundant storage</a> )	<a href="#">Azure Backup</a>
Managed disks	Local ( <a href="#">locally redundant storage</a> )	<a href="#">Azure Backup</a>
Unmanaged locally redundant storage disks	Local ( <a href="#">locally redundant storage</a> )	<a href="#">Azure Backup</a>
Unmanaged geo-redundant storage disks	Cross region ( <a href="#">geo-redundant storage</a> )	<a href="#">Azure Backup</a> <a href="#">Consistent snapshots</a>
Unmanaged read-access geo-redundant storage disks	Cross region ( <a href="#">read-access geo-redundant storage</a> )	<a href="#">Azure Backup</a> <a href="#">Consistent snapshots</a>

High availability is best met by using managed disks in an availability set along with Azure Backup. If you use unmanaged disks, you can still use Azure Backup for DR. If you are unable to use Azure Backup, then taking [consistent snapshots](#), as described in a later section, is an alternative solution for backup and DR.

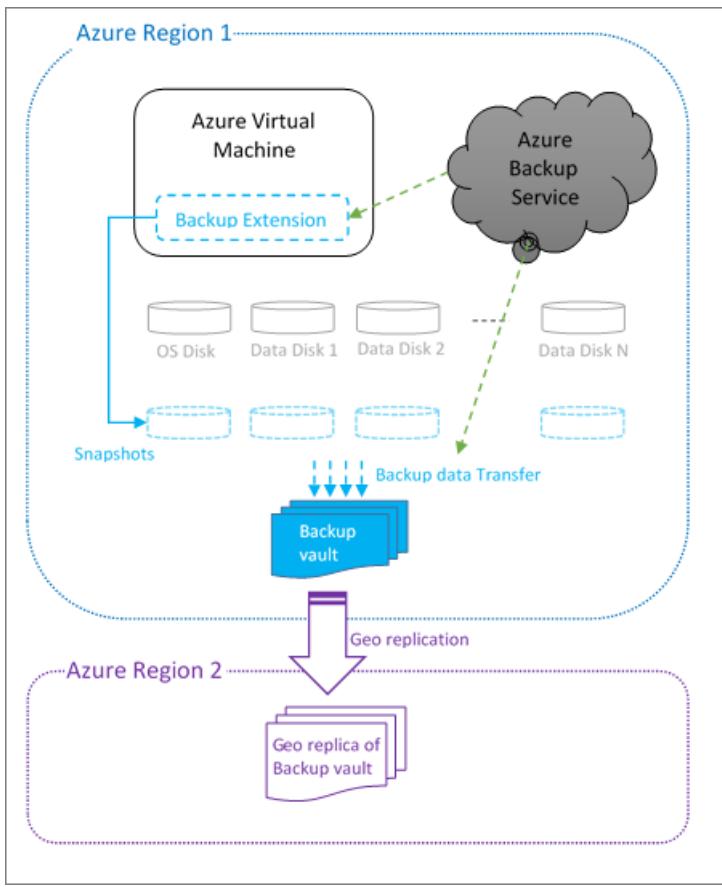
Your choices for high availability, backup, and DR at application or infrastructure levels can be represented as follows:

LEVEL	HIGH AVAILABILITY	BACKUP OR DR
Application	SQL Server Always On	<a href="#">Azure Backup</a>
Infrastructure	Availability set	Geo-redundant storage with consistent snapshots

## Using Azure Backup

[Azure Backup](#) can back up your VMs running Windows or Linux to the Azure recovery services vault. Backing up and restoring business-critical data is complicated by the fact that business-critical data must be backed up while the applications that produce the data are running.

To address this issue, Azure Backup provides application-consistent backups for Microsoft workloads. It uses the volume shadow service to ensure that data is written correctly to storage. For Linux VMs, the default backup consistency mode is file-consistent backups, because Linux does not have functionality equivalent to the volume shadow service as in the case of Windows. For Linux machines, see [Application-consistent backup of Azure Linux VMs](#).



When Azure Backup initiates a backup job at the scheduled time, it triggers the backup extension installed in the VM to take a point-in-time snapshot. A snapshot is taken in coordination with the volume shadow service to get a consistent snapshot of the disks in the virtual machine without having to shut it down. The backup extension in the VM flushes all writes before taking a consistent snapshot of all of the disks. After taking the snapshot, the data is transferred by Azure Backup to the backup vault. To make the backup process more efficient, the service identifies and transfers only the blocks of data that have changed after the last backup.

To restore, you can view the available backups through Azure Backup and then initiate a restore. You can create and restore Azure backups through the [Azure portal](#), by [using PowerShell](#), or by using the [Azure CLI](#).

### Steps to enable a backup

Use the following steps to enable backups of your VMs by using the [Azure portal](#). There is some variation depending on your exact scenario. Refer to the [Azure Backup](#) documentation for full details. Azure Backup also [supports VMs with managed disks](#).

1. Create a recovery services vault for a VM:
  - a. In the [Azure portal](#), browse All resources and find **Recovery Services vaults**.
  - b. On the **Recovery Services vaults** menu, click **Add** and follow the steps to create a new vault in the same region as the VM. For example, if your VM is in the West US region, pick West US for the vault.
2. Verify the storage replication for the newly created vault. Access the vault under **Recovery Services vaults** and go to **Properties > Backup Configuration > Update**. Ensure the **geo-redundant storage** option is selected by default. This option ensures that your vault is automatically replicated to a secondary datacenter. For example, your vault in West US is automatically replicated to East US.
3. Configure the backup policy and select the VM from the same UI.
4. Make sure the Backup Agent is installed on the VM. If your VM is created by using an Azure gallery image, then the Backup Agent is already installed. Otherwise (that is, if you use a custom image), use the instructions to [install the VM agent on a virtual machine](#).

5. After the previous steps are completed, the backup runs at regular intervals as specified in the backup policy. If necessary, you can trigger the first backup manually from the vault dashboard on the Azure portal.

For automating Azure Backup by using scripts, refer to [PowerShell cmdlets for VM backup](#).

### Steps for recovery

If you need to repair or rebuild a VM, you can restore the VM from any of the backup recovery points in the vault. There are a couple of different options for performing the recovery:

- You can create a new VM as a point-in-time representation of your backed-up VM.
- You can restore the disks, and then use the template for the VM to customize and rebuild the restored VM.

For more information, see the instructions to [use the Azure portal to restore virtual machines](#). This document also explains the specific steps for restoring backed-up VMs to a paired datacenter by using your geo-redundant backup vault if there is a disaster at the primary datacenter. In that case, Azure Backup uses the Compute service from the secondary region to create the restored virtual machine.

You can also use PowerShell for [creating a new VM from restored disks](#).

## Alternative solution: Consistent snapshots

If you are unable to use Azure Backup, you can implement your own backup mechanism by using snapshots. Creating consistent snapshots for all the disks used by a VM and then replicating those snapshots to another region is complicated. For this reason, Azure considers using the Backup service as a better option than building a custom solution.

If you use read-access geo-redundant storage/geo-redundant storage for disks, snapshots are automatically replicated to a secondary datacenter. If you use locally redundant storage for disks, you need to replicate the data yourself. For more information, see [Back up Azure-unmanaged VM disks with incremental snapshots](#).

A snapshot is a representation of an object at a specific point in time. A snapshot incurs billing for the incremental size of the data it holds. For more information, see [Create a blob snapshot](#).

### Create snapshots while the VM is running

Although you can take a snapshot at any time, if the VM is running, there is still data being streamed to the disks. The snapshots might contain partial operations that were in flight. Also, if there are several disks involved, the snapshots of different disks might have occurred at different times. These scenarios may cause the snapshots to be uncoordinated. This lack of coordination is especially problematic for striped volumes whose files might be corrupted if changes were being made during backup.

To avoid this situation, the backup process must implement the following steps:

1. Freeze all the disks.
2. Flush all the pending writes.
3. [Create a blob snapshot](#) for all the disks.

Some Windows applications, like SQL Server, provide a coordinated backup mechanism via a volume shadow service to create application-consistent backups. On Linux, you can use a tool like `fsfreeze` for coordinating the disks. This tool provides file-consistent backups, but not application-consistent snapshots. This process is complex, so you should consider using [Azure Backup](#) or a third-party backup solution that already implements this procedure.

The previous process results in a collection of coordinated snapshots for all of the VM disks, representing a

specific point-in-time view of the VM. This is a backup restore point for the VM. You can repeat the process at scheduled intervals to create periodic backups. See [Copy the backups to another region](#) for steps to copy the snapshots to another region for DR.

### Create snapshots while the VM is offline

Another option to create consistent backups is to shut down the VM and take blob snapshots of each disk. Taking blob snapshots is easier than coordinating snapshots of a running VM, but it requires a few minutes of downtime.

1. Shut down the VM.
2. Create a snapshot of each virtual hard drive blob, which only takes a few seconds.

To create a snapshot, you can use [PowerShell](#), the [Azure Storage REST API](#), [Azure CLI](#), or one of the Azure Storage client libraries, such as [the Storage client library for .NET](#).

3. Start the VM, which ends the downtime. Typically, the entire process finishes within a few minutes.

This process yields a collection of consistent snapshots for all the disks, providing a backup restore point for the VM.

### Copy the snapshots to another region

Creation of the snapshots alone might not be sufficient for DR. You must also replicate the snapshot backups to another region.

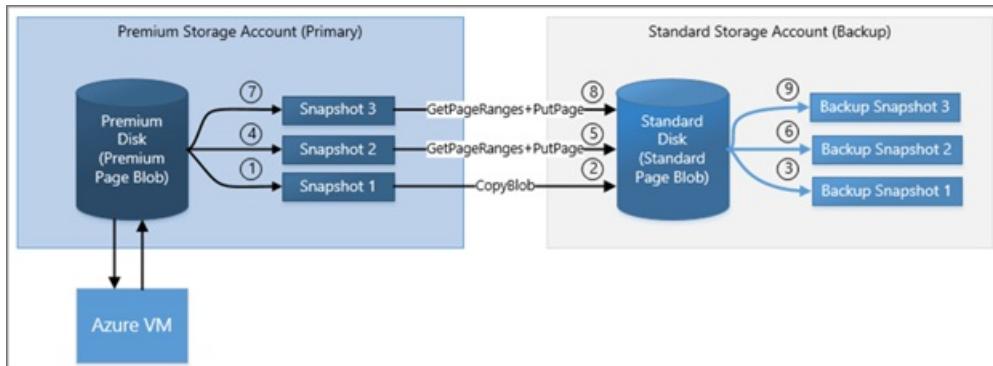
If you use geo-redundant storage or read-access geo-redundant storage for your disks, then the snapshots are replicated to the secondary region automatically. There can be a few minutes of lag before the replication. If the primary datacenter goes down before the snapshots finish replicating, you cannot access the snapshots from the secondary datacenter. The likelihood of this is small.

#### NOTE

Only having the disks in a geo-redundant storage or read-access geo-redundant storage account does not protect the VM from disasters. You must also create coordinated snapshots or use Azure Backup. This is required to recover a VM to a consistent state.

If you use locally redundant storage, you must copy the snapshots to a different storage account immediately after creating the snapshot. The copy target might be a locally redundant storage account in a different region, resulting in the copy being in a remote region. You can also copy the snapshot to a read-access geo-redundant storage account in the same region. In this case, the snapshot is lazily replicated to the remote secondary region. Your backup is protected from disasters at the primary site after the copying and replication is complete.

To copy your incremental snapshots for DR efficiently, review the instructions in [Back up Azure unmanaged VM disks with incremental snapshots](#).



### Recovery from snapshots

To retrieve a snapshot, copy it to make a new blob. If you are copying the snapshot from the primary account, you can copy the snapshot over to the base blob of the snapshot. This process reverts the disk to the snapshot. This process is known as promoting the snapshot. If you are copying the snapshot backup from a secondary account, in the case of a read-access geo-redundant storage account, you must copy it to a primary account. You can copy a snapshot by [using PowerShell](#) or by using the AzCopy utility. For more information, see [Transfer data with the AzCopy command-line utility](#).

For VMs with multiple disks, you must copy all the snapshots that are part of the same coordinated restore point. After you copy the snapshots to writable VHD blobs, you can use the blobs to recreate your VM by using the template for the VM.

## Other options

### SQL Server

SQL Server running in a VM has its own built-in capabilities to back up your SQL Server database to Azure Blob storage or a file share. If the storage account is geo-redundant storage or read-access geo-redundant storage, you can access those backups in the storage account's secondary datacenter in the event of a disaster, with the same restrictions as previously discussed. For more information, see [Back up and restore for SQL Server in Azure virtual machines](#). In addition to back up and restore, [SQL Server Always On availability groups](#) can maintain secondary replicas of databases. This ability greatly reduces the disaster recovery time.

## Other considerations

This article has discussed how to back up or take snapshots of your VMs and their disks to support disaster recovery and how to use those backups or snapshots to recover your data. With the Azure Resource Manager model, many people use templates to create their VMs and other infrastructures in Azure. You can use a template to create a VM that has the same configuration every time. If you use custom images for creating your VMs, you must also make sure that your images are protected by using a read-access geo-redundant storage account to store them.

Consequently, your backup process can be a combination of two things:

- Back up the data (disks).
- Back up the configuration (templates and custom images).

Depending on the backup option you choose, you might have to handle the backup of both the data and the configuration, or the backup service might handle all of that for you.

## Appendix: Understanding the impact of data redundancy

For storage accounts in Azure, there are three types of data redundancy that you should consider regarding disaster recovery: locally redundant, geo-redundant, or geo-redundant with read access.

Locally redundant storage retains three copies of the data in the same datacenter. When the VM writes the data, all three copies are updated before success is returned to the caller, so you know they are identical. Your disk is protected from local failures, because it's unlikely that all three copies are affected at the same time. In the case of locally redundant storage, there is no geo-redundancy, so the disk is not protected from catastrophic failures that can affect an entire datacenter or storage unit.

With geo-redundant storage and read-access geo-redundant storage, three copies of your data are retained in the primary region that is selected by you. Three more copies of your data are retained in a corresponding secondary region that is set by Azure. For example, if you store data in West US, the data is replicated to East US. Copy retention is done asynchronously, and there is a small delay between updates to the primary and secondary sites. Replicas of the disks on the secondary site are consistent on a per-disk basis (with the delay), but replicas of multiple active disks might not be in sync with each other. To have consistent replicas across

multiple disks, consistent snapshots are needed.

The main difference between geo-redundant storage and read-access geo-redundant storage is that with read-access geo-redundant storage, you can read the secondary copy at any time. If there is a problem that renders the data in the primary region inaccessible, the Azure team makes every effort to restore access. While the primary is down, if you have read-access geo-redundant storage enabled, you can access the data in the secondary datacenter. Therefore, if you plan to read from the replica while the primary is inaccessible, then read-access geo-redundant storage should be considered.

If it turns out to be a significant outage, the Azure team might trigger a geo-failover and change the primary DNS entries to point to secondary storage. At this point, if you have either geo-redundant storage or read-access geo-redundant storage enabled, you can access the data in the region that used to be the secondary. In other words, if your storage account is geo-redundant storage and there is a problem, you can access the secondary storage only if there is a geo-failover.

For more information, see [What to do if an Azure Storage outage occurs](#).

## Next steps

See [Back up Azure unmanaged Virtual Machine disks with incremental snapshots](#).

# Ephemeral OS disks for Azure VMs

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Ephemeral OS disks are created on the local virtual machine (VM) storage and not saved to the remote Azure Storage. Ephemeral OS disks work well for stateless workloads, where applications are tolerant of individual VM failures but are more affected by VM deployment time or reimaging of individual VM instances. With Ephemeral OS disk, you get lower read/write latency to the OS disk and faster VM reimage.

The key features of ephemeral disks are:

- Ideal for stateless applications.
- Supported by Marketplace, custom images, and by [Azure Compute Gallery](#) (formerly known as Shared Image Gallery).
- Ability to fast reset or reimage VMs and scale set instances to the original boot state.
- Lower latency, similar to a temporary disk.
- Ephemeral OS disks are free, you incur no storage cost for OS disks.
- Available in all Azure regions.

Key differences between persistent and ephemeral OS disks:

	PERSISTENT OS DISK	EPHEMERAL OS DISK
Size limit for OS disk	2 TiB	Cache size or temp size for the VM size or 2040 GiB, whichever is smaller. For the <b>cache or temp size in GiB</b> , see <a href="#">DS</a> , <a href="#">ES</a> , <a href="#">M</a> , <a href="#">FS</a> , and <a href="#">GS</a>
VM sizes supported	All	VM sizes that support Premium storage such as DSv1, DSv2, DSv3, Esv3, Fs, FsV2, GS, M, Mdsv2, Bs, Dav4, Eav4
Disk type support	Managed and unmanaged OS disk	Managed OS disk only
Region support	All regions	All regions
Data persistence	OS disk data written to OS disk are stored in Azure Storage	Data written to OS disk is stored on local VM storage and isn't persisted to Azure Storage.
Stop-deallocated state	VMs and scale set instances can be stop-deallocated and restarted from the stop-deallocated state	Not Supported
Specialized OS disk support	Yes	No
OS disk resize	Supported during VM creation and after VM is stop-deallocated	Supported during VM creation only

	PERSISTENT OS DISK	EPHEMERAL OS DISK
Resizing to a new VM size	OS disk data is preserved	Data on the OS disk is deleted, OS is reprovisioned
Redeploy	OS disk data is preserved	Data on the OS disk is deleted, OS is reprovisioned
Stop/ Start of VM	OS disk data is preserved	Not Supported
Page file placement	For Windows, page file is stored on the resource disk	For Windows, page file is stored on the OS disk (for both OS cache placement and Temp disk placement).
Maintenance of VM/VMSS using <b>healing</b>	OS disk data is preserved	OS disk data is not preserved
Maintenance of VM/VMSS using <b>Live Migration</b>	OS disk data is preserved	OS disk data is preserved

## Placement options for Ephemeral OS disks

Ephemeral OS disk can be stored either on VM's OS cache disk or VM's temp/resource disk. [DiffDiskPlacement](#) is the new property that can be used to specify where you want to place the Ephemeral OS disk. With this feature, when a Windows VM is provisioned, we configure the pagefile to be located on the OS Disk.

## Size requirements

You can choose to deploy Ephemeral OS Disk on VM cache or VM temp disk. The image OS disk's size should be less than or equal to the temp/cache size of the VM size chosen.

For example, if you want to opt for **OS cache placement**: Standard Windows Server images from the marketplace are about 127 GiB, which means that you need a VM size that has a cache equal to or larger than 127 GiB. The Standard\_DS3\_v2 has a cache size of 127 GiB, which is large enough. In this case, the Standard\_DS3\_v2 is the smallest size in the DSv2 series that you can use with this image.

If you want to opt for **Temp disk placement**: Standard Ubuntu server image from marketplace is about 30 GiB. To enable Ephemeral OS disk on temp, the temp disk size must be equal to or larger than 30 GiB. Standard\_B4ms has a temp size of 32 GiB, which can fit the 30 GiB OS disk. Upon creation of the VM, the temp disk space would be 2 GiB.

### IMPORTANT

If opting for temp disk placement the Final Temp disk size = (Initial temp disk size - OS image size).

In the case of **Temp disk placement**, as Ephemeral OS disk is placed on temp disk it will share the IOPS with temp disk as per the VM size chosen by you.

Basic Linux and Windows Server images in the Marketplace that are denoted by `[smallsize]` tend to be around 30 GiB and can use most of the available VM sizes. Ephemeral disks also require that the VM size supports **Premium storage**. The sizes usually (but not always) have an `s` in the name, like DSv2 and EsV3. For more information, see [Azure VM sizes](#) for details around which sizes support Premium storage.

#### NOTE

Ephemeral disk will not be accessible through the portal. You will receive a "Resource not Found" or "404" error when accessing the ephemeral disk which is expected.

## Unsupported features

- Capturing VM images
- Disk snapshots
- Azure Disk Encryption
- Azure Backup
- Azure Site Recovery
- OS Disk Swap

## Trusted Launch for Ephemeral OS disks

Ephemeral OS disks can be created with Trusted launch. Not all VM sizes and regions are supported for trusted launch. Check [limitations of trusted launch](#) for supported sizes and regions. VM guest state (VMGS) is specific to trusted launch VMs. It is a blob that is managed by Azure and contains the unified extensible firmware interface (UEFI) secure boot signature databases and other security information. When using trusted launch by default 1 GiB from the **OS cache or temp storage** based on the chosen placement option is reserved for VMGS. The lifecycle of the VMGS blob is tied to that of the OS Disk.

For example, If you try to create a Trusted launch Ephemeral OS disk VM using OS image of size 56 GiB with VM size [Standard\\_DS4\\_v2](#) using temp disk placement you would get an error as "**OS disk of Ephemeral VM with size greater than 55 GB is not allowed for VM size Standard\_DS4\_v2 when the DiffDiskPlacement is ResourceDisk.**" This is because the temp storage for [Standard\\_DS4\\_v2](#) is 56 GiB, and 1 GiB is reserved for VMGS when using trusted launch. For the same example above, if you create a standard Ephemeral OS disk VM you would not get any errors and it would be a successful operation.

#### IMPORTANT

While using ephemeral disks for Trusted Launch VMs, keys and secrets generated or sealed by the vTPM after VM creation may not be persisted for operations like reimaging and platform events like service healing.

For more information on [how to deploy a trusted launch VM](#)

## Confidential VMs using Ephemeral OS disks (preview)

AMD-based Confidential VMs cater to high security and confidentiality requirements of customers. These VMs provide a strong, hardware-enforced boundary to help meet your security needs. There are limitations to use Confidential VMs. Check the [region](#), [size](#) and [OS supported](#) limitations for confidential VMs. Virtual machine guest state (VMGS) blob contains the security information of the confidential VM. Confidential VMs using Ephemeral OS disks by default 1 GiB from the **OS cache or temp storage** based on the chosen placement option is reserved for VMGS. The lifecycle of the VMGS blob is tied to that of the OS Disk.

#### IMPORTANT

When choosing a confidential VM with full OS disk encryption before VM deployment that uses a customer-managed key (CMK). [Updating a CMK key version](#) or [key rotation](#) is not supported with Ephemeral OS disk. Confidential VMs using Ephemeral OS disks need to be deleted before updating or rotating the keys and can be re-created subsequently.

For more information on [confidential VM](#)

## Customer Managed key (preview)

You can choose to use customer managed keys or platform managed keys when you enable end-to-end encryption for VMs using Ephemeral OS disk. Currently this option is available only via [PowerShell](#), [CLI](#) and SDK in all regions.

### IMPORTANT

Updating a CMK key version or key rotation of customer managed key is not supported with Ephemeral OS disk. VMs using Ephemeral OS disks need to be deleted before updating or rotating the keys and can be re-created subsequently.

For more information on [Encryption at host](#)

## Next steps

Create a VM with ephemeral OS disk using [Azure Portal/CLI/PowerShell/ARM template](#). Check out the [frequently asked questions on ephemeral os disk](#).

# How to deploy Ephemeral OS disks for Azure VMs

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article shows you how to create a virtual machine or virtual machine scale sets with Ephemeral OS disks through Portal, ARM template deployment, CLI and PowerShell.

## Portal

In the Azure portal, you can choose to use ephemeral disks when deploying a virtual machine or virtual machine scale sets by opening the **Advanced** section of the **Disks** tab. For choosing placement of Ephemeral OS disk, select **OS cache placement** or **Temp disk placement**.

Home >

## Create a virtual machine

Basics Disks Networking Management Advanced Tags Review + create

Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. [Learn more](#)

Disk options

OS disk type \* ⓘ Premium SSD (locally-redundant storage)

Encryption type \* ⓘ (Default) Encryption at-rest with a platform-managed key

Enable Ultra Disk compatibility ⓘ

Data disks

You can add and configure additional data disks for your virtual machine or attach existing disks. This VM also comes with a temporary disk.

LUN	Name	Size (GiB)	Disk type	Host caching
-----	------	------------	-----------	--------------

[Create and attach a new disk](#) [Attach an existing disk](#)

Advanced

Use managed disks ⓘ  Availability zone requires managed disks.

Ephemeral OS disk ⓘ  None  OS cache placement  Temp disk placement

[Review + create](#) [< Previous](#) [Next : Networking >](#)

If the option for using an ephemeral disk or OS cache placement or Temp disk placement is greyed out, you might have selected a VM size that doesn't have a cache/temp size larger than the OS image or that doesn't support Premium storage. Go back to the **Basics** page and try choosing another VM size.

## Scale set template deployment

The process to create a scale set that uses an ephemeral OS disk is to add the `diffDiskSettings` property to the `Microsoft.Compute/virtualMachineScaleSets/virtualMachineProfile` resource type in the template. Also, the caching policy must be set to `ReadOnly` for the ephemeral OS disk. placement can be changed to `CacheDisk` for OS cache disk placement.

```
{  
  "type": "Microsoft.Compute/virtualMachineScaleSets",  
  "name": "myScaleSet",  
  "location": "East US 2",  
  "apiVersion": "2019-12-01",  
  "sku": {  
    "name": "Standard_DS2_v2",  
    "capacity": "2"  
  },  
  "properties": {  
    "upgradePolicy": {  
      "mode": "Automatic"  
    },  
    "virtualMachineProfile": {  
      "storageProfile": {  
        "osDisk": {  
          "diffDiskSettings": {  
            "option": "Local" ,  
            "placement": "ResourceDisk"  
          },  
          "caching": "ReadOnly",  
          "createOption": "FromImage"  
        },  
        "imageReference": {  
          "publisher": "Canonical",  
          "offer": "UbuntuServer",  
          "sku": "16.04-LTS",  
          "version": "latest"  
        }  
      },  
      "osProfile": {  
        "computerNamePrefix": "myvmss",  
        "adminUsername": "azureuser",  
        "adminPassword": "P@ssw0rd!"  
      }  
    }  
  }  
}
```

## VM template deployment

You can deploy a VM with an ephemeral OS disk using a template. The process to create a VM that uses ephemeral OS disks is to add the `diffDiskSettings` property to `Microsoft.Compute/virtualMachines` resource type in the template. Also, the caching policy must be set to `ReadOnly` for the ephemeral OS disk. placement option can be changed to `CacheDisk` for OS cache disk placement.

```
{
  "type": "Microsoft.Compute/virtualMachines",
  "name": "myVirtualMachine",
  "location": "East US 2",
  "apiVersion": "2019-12-01",
  "properties": {
    "storageProfile": {
      "osDisk": {
        "diffDiskSettings": {
          "option": "Local",
          "placement": "ResourceDisk"
        },
        "caching": "ReadOnly",
        "createOption": "FromImage"
      },
      "imageReference": {
        "publisher": "MicrosoftWindowsServer",
        "offer": "WindowsServer",
        "sku": "2016-Datacenter-smalldisk",
        "version": "latest"
      },
      "hardwareProfile": {
        "vmSize": "Standard_DS2_v2"
      }
    },
    "osProfile": {
      "computerNamePrefix": "myvirtualmachine",
      "adminUsername": "azureuser",
      "adminPassword": "P@ssw0rd!"
    }
  }
}
```

## CLI

To use an ephemeral disk for a CLI VM deployment, set the `--ephemeral-os-disk` parameter in `az vm create` to `true` and the `--ephemeral-os-disk-placement` parameter to `ResourceDisk` for temp disk placement or `CacheDisk` for cache disk placement and the `--os-disk-caching` parameter to `ReadOnly`.

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image UbuntuLTS \
--ephemeral-os-disk true \
--ephemeral-os-disk-placement ResourceDisk \
--os-disk-caching ReadOnly \
--admin-username azureuser \
--generate-ssh-keys
```

For scale sets, you use the same `--ephemeral-os-disk true` parameter for `az-vmss-create` and set the `--os-disk-caching` parameter to `ReadOnly` and the `--ephemeral-os-disk-placement` parameter to `ResourceDisk` for temp disk placement or `CacheDisk` for cache disk placement.

## Reimage a VM using REST

You can reimagine a Virtual Machine instance with ephemeral OS disk using REST API as described below and via Azure portal by going to Overview pane of the VM. For scale sets, reimaging is already available through PowerShell, CLI, and the portal.

```
POST https://management.azure.com/subscriptions/{sub-  
id}/resourceGroups/{rgName}/providers/Microsoft.Compute/VirtualMachines/{vmName}/reimage?api-version=2019-  
12-01"
```

## PowerShell

To use an ephemeral disk for a PowerShell VM deployment, use [Set-AzVMOSDisk](#) in your VM configuration. Set the `-DiffDiskSetting` to `Local` and `-Caching` to `ReadOnly` and `-DiffDiskPlacement` to `ResourceDisk`.

```
Set-AzVMOSDisk -DiffDiskSetting Local -DiffDiskPlacement ResourceDisk -Caching ReadOnly
```

To use an ephemeral disk on cache disk for a PowerShell VM deployment, use [Set-AzVMOSDisk](#) in your VM configuration. Set the `-DiffDiskSetting` to `Local` , `-Caching` to `ReadOnly` and `-DiffDiskPlacement` to `CacheDisk` .

```
Set-AzVMOSDisk -DiffDiskSetting Local -DiffDiskPlacement CacheDisk -Caching ReadOnly
```

For scale set deployments, use the [Set-AzVmssStorageProfile](#) cmdlet in your configuration. Set the `-DiffDiskSetting` to `Local` , `-Caching` to `ReadOnly` and `-DiffDiskPlacement` to `ResourceDisk` or `CacheDisk` .

```
Set-AzVmssStorageProfile -DiffDiskSetting Local -DiffDiskPlacement ResourceDisk -OsDiskCaching ReadOnly
```

## Next steps

For more information on [Ephemeral OS disk](#).

# Frequently asked questions about Ephemeral OS disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

**Q: What is the size of the local OS Disks?**

A: We support platform, Shared Image Gallery, and custom images, up to the VM cache size with OS cache placement and up to Temp disk size with Temp disk placement, where all read/writes to the OS disk will be local on the same node as the Virtual Machine.

**Q: Can the ephemeral OS disk be resized?**

A: No, once the ephemeral OS disk is provisioned, the OS disk cannot be resized.

**Q: Can the ephemeral OS disk placement be modified after creation of VM?**

A: No, once the ephemeral OS disk is provisioned, the OS disk placement cannot be changed. But the VM can be recreated via ARM template deployment/PowerShell/CLI by updating the OS disk placement of choosing. This would result in the recreation of the VM with Data on the OS disk deleted and OS is reprovisioned.

**Q: Is there any Temp disk created if image size equals to Temp disk size of VM size selected?**

A: No, in that case, there won't be any Temp disk drive created.

**Q: Are Ephemeral OS disks supported on low-priority VMs and Spot VMs?**

A: Yes. There is no option of Stop-Deallocate for Ephemeral VMs, rather users need to Delete instead of deallocating them.

**Q: Can I attach a Managed Disks to an Ephemeral VM?**

A: Yes, you can attach a managed data disk to a VM that uses an ephemeral OS disk.

**Q: Will all VM sizes be supported for ephemeral OS disks?**

A: No, most Premium Storage VM sizes are supported (DS, ES, FS, GS, M, etc.). To know whether a particular VM size supports ephemeral OS disks for an OS image size you can use the below script. It takes the OS image size and location as inputs and provides a list of VM SKUs and corresponding placement supported. If both OS Cache and temp disk placement are marked as not supported then Ephemeral OS disk cannot be used for the given OS image size.

```

[CmdletBinding()]
param([Parameter(Mandatory=$true)]
    [ValidateNotNullOrEmpty()])
    [string]$Location,
    [Parameter(Mandatory=$true)]
    [long]$OSImageSizeInGB
)

Function HasSupportEphemeralOSDisk([object[]] $capability)
{
    return $capability | where { $_.Name -eq "EphemeralOSDiskSupported" -and $_.Value -eq "True" }
}

Function Get-MaxTempDiskAndCacheSize([object[]] $capabilities)
{
    $MaxResourceVolumeGB = 0;
    $CachedDiskGB = 0;

    foreach($capability in $capabilities)
    {
        if ($capability.Name -eq "MaxResourceVolumeMB")
        { $MaxResourceVolumeGB = [int]($capability.Value / 1024) }

        if ($capability.Name -eq "CachedDiskBytes")
        { $CachedDiskGB = [int]($capability.Value / (1024 * 1024 * 1024)) }
    }

    return ($MaxResourceVolumeGB, $CachedDiskGB)
}

Function Get-EphemeralSupportedVMSku
{
    [CmdletBinding()]
    Param
    (
        [Parameter(Mandatory=$true)]
        [long]$OSImageSizeInGB,
        [Parameter(Mandatory=$true)]
        [string]$Location
    )

    $VmSkus = Get-AzComputeResourceSku $Location | Where-Object { $_.ResourceType -eq "virtualMachines" -and (HasSupportEphemeralOSDisk $_.Capabilities) -ne $null }

    $Response = @()
    foreach ($sku in $VmSkus)
    {
        ($MaxResourceVolumeGB, $CachedDiskGB) = Get-MaxTempDiskAndCacheSize $sku.Capabilities

        $Response += New-Object PSObject -Property @{
            ResourceSKU = $sku.Size
            TempDiskPlacement = @{ $true = "NOT SUPPORTED"; $false = "SUPPORTED"}[$MaxResourceVolumeGB -lt $OSImageSizeInGB]
            CacheDiskPlacement = @{ $true = "NOT SUPPORTED"; $false = "SUPPORTED"}[$CachedDiskGB -lt $OSImageSizeInGB]
        };
    }

    return $Response
}

Get-EphemeralSupportedVMSku -OSImageSizeInGB $OSImageSizeInGB -Location $Location | Format-Table

```

**Q: Can the ephemeral OS disk be applied to existing VMs and scale sets?**

A: No, ephemeral OS disk can only be used during VM and scale set creation.

**Q: Can you mix ephemeral and normal OS disks in a scale set?**

A: No, you can't have a mix of ephemeral and persistent OS disk instances within the same scale set.

**Q: Can the ephemeral OS disk be created using PowerShell or CLI?**

A: Yes, you can create VMs with Ephemeral OS Disk using REST, Templates, PowerShell, and CLI.

# Azure CLI - Restrict import/export access for managed disks with Private Links

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You can use [private endpoints](#) to restrict the export and import of managed disks and securely access data over a [Private Link](#) from clients on your Azure virtual network. The private endpoint uses an IP address from the virtual network address space for your managed disks service. Network traffic between clients on their virtual network and managed disks only traverses over the virtual network and a private link on the Microsoft backbone network, eliminating exposure from the public internet.

To use Private Links to export/import managed disks, first you create a disk access resource and link it to a virtual network in the same subscription by creating a private endpoint. Then, associate a disk or a snapshot with an instance of disk access. Finally, set the `NetworkAccessPolicy` property of the disk or the snapshot to `AllowPrivate`. This will limit access to your virtual network.

You can set the `NetworkAccessPolicy` property to `DenyAll` to prevent anybody from exporting data of a disk or a snapshot. The default value for the `NetworkAccessPolicy` property is `AllowAll`.

## Limitations

- Your virtual network must be in the same subscription as your disk access object to link them.
- You cannot import or export more than 10 disks or snapshots at the same time with the same disk access object.
- You cannot request manual approval to link a virtual network to a disk access object.

## Log in into your subscription and set your variables

```
subscriptionId=yourSubscriptionId
resourceGroupName=yourResourceGroupName
region=northcentralus
diskAccessName=yourDiskAccessForPrivateLinks
vnetName=yourVNETForPrivateLinks
subnetName=yourSubnetForPrivateLinks
privateEndPointName=yourPrivateLinkForSecureMDExportImport
privateEndPointConnectionName=yourPrivateLinkConnection

#The name of an existing disk which is the source of the snapshot
sourceDiskName=yourSourceDiskForSnapshot

#The name of the new snapshot which will be secured via Private Links
snapshotNameSecuredWithPL=yourSnapshotNameSecuredWithPL

az login

az account set --subscription $subscriptionId
```

## Create a disk access using Azure CLI

```
az disk-access create -n $diskAccessName -g $resourceGroupName -l $region  
diskAccessId=$(az disk-access show -n $diskAccessName -g $resourceGroupName --query [id] -o tsv)
```

## Create a Virtual Network

Network policies like network security groups (NSG) are not supported for private endpoints. In order to deploy a Private Endpoint on a given subnet, an explicit disable setting is required on that subnet.

```
az network vnet create --resource-group $resourceGroupName \  
--name $vnetName \  
--subnet-name $subnetName
```

## Disable subnet private endpoint policies

Azure deploys resources to a subnet within a virtual network, so you need to update the subnet to disable private endpoint network policies.

```
az network vnet subnet update --resource-group $resourceGroupName \  
--name $subnetName \  
--vnet-name $vnetName \  
--disable-private-endpoint-network-policies true
```

## Create a private endpoint for the disk access object

```
az network private-endpoint create --resource-group $resourceGroupName \  
--name $privateEndPointName \  
--vnet-name $vnetName \  
--subnet $subnetName \  
--private-connection-resource-id $diskAccessId \  
--group-ids disks \  
--connection-name $privateEndPointConnectionName
```

## Configure the Private DNS Zone

Create a Private DNS Zone for Storage blob domain, create an association link with the Virtual Network and create a DNS Zone Group to associate the private endpoint with the Private DNS Zone.

```
az network private-dns zone create --resource-group $resourceGroupName \  
--name "privatelink.blob.core.windows.net"  
  
az network private-dns link vnet create --resource-group $resourceGroupName \  
--zone-name "privatelink.blob.core.windows.net" \  
--name yourDNSLink \  
--virtual-network $vnetName \  
--registration-enabled false  
  
az network private-endpoint dns-zone-group create \  
--resource-group $resourceGroupName \  
--endpoint-name $privateEndPointName \  
--name yourZoneGroup \  
--private-dns-zone "privatelink.blob.core.windows.net" \  
--zone-name disks
```

## Create a disk protected with Private Links

```
resourceGroupName=yourResourceGroupName
region=northcentralus
diskAccessName=yourDiskAccessName
diskName=yourDiskName
diskSkuName=Standard_LRS
diskSizeGB=128

diskAccessId=$(az resource show -n $diskAccessName -g $resourceGroupName --namespace Microsoft.Compute --
resource-type diskAccesses --query [id] -o tsv)

az disk create -n $diskName \
-g $resourceGroupName \
-l $region \
--size-gb $diskSizeGB \
--sku $diskSkuName \
--network-access-policy AllowPrivate \
--disk-access $diskAccessId
```

## Create a snapshot of a disk protected with Private Links

```
resourceGroupName=yourResourceGroupName
region=northcentralus
diskAccessName=yourDiskAccessName
sourceDiskName=yourSourceDiskForSnapshot
snapshotNameSecuredWithPL=yourSnapshotName

diskId=$(az disk show -n $sourceDiskName -g $resourceGroupName --query [id] -o tsv)

diskAccessId=$(az resource show -n $diskAccessName -g $resourceGroupName --namespace Microsoft.Compute --
resource-type diskAccesses --query [id] -o tsv)

az snapshot create -n $snapshotNameSecuredWithPL \
-g $resourceGroupName \
-l $region \
--source $diskId \
--network-access-policy AllowPrivate \
--disk-access $diskAccessId
```

## Next steps

- Upload a VHD to Azure or copy a managed disk to another region - [Azure CLI](#) or [Azure PowerShell module](#)
- Download a VHD - [Windows](#) or [Linux](#)
- [FAQ on Private Links](#)
- [Export/Copy managed snapshots as VHD to a storage account in different region with CLI](#)

# Restrict import/export access for managed disks using Azure Private Link

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can use [private endpoints](#) to restrict the export and import of managed disks and more securely access data over a [private link](#) from clients on your Azure virtual network. The private endpoint uses an IP address from the virtual network address space for your managed disks. Network traffic between clients on their virtual network and managed disks only traverses over the virtual network and a private link on the Microsoft backbone network, eliminating exposure from the public internet.

To use Private Link to export and import managed disks, first you create a disk access resource and link it to a virtual network in the same subscription by creating a private endpoint. Then, associate a disk or a snapshot with a disk access instance.

## Limitations

- Your virtual network must be in the same subscription as your disk access object to link them.
- You cannot import or export more than 10 disks or snapshots at the same time with the same disk access object.
- You cannot request manual approval to link a virtual network to a disk access object.

## Create a disk access resource

1. Sign in to the Azure portal and navigate to **Disk Accesses** with [this link](#).

### IMPORTANT

You must use the [provided link](#) to navigate to the Disk Accesses pane. It is not currently visible in the public portal without using the link.

2. Select **+ Create** to create a new disk access resource.
3. On the **Create a disk access** pane, select your subscription and a resource group. Under **Instance details**, enter a name and select a region.

## Create a disk access ...

Basics Tags Review + create

Private links provide protection from a SAS URI being available to anyone by locking down the access to a specific virtual network. This private "tunnel" gives users the security they need when sensitive data is contained on the disks or snapshots.

[Learn more](#)

### Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

My Example Subscription



Resource group \* ⓘ

myResourceGroup



[Create new](#)

### Instance details

Name \* ⓘ

example-disk-access-name



Region \* ⓘ

(US) West US 2



### Private endpoint

Name	Subscription	Resource group	Region	Target resource
<i>Click on add to create a private endpoint</i>				
<a href="#">+ Add</a>				

4. Select **Review + create**.

5. When your resource has been created, navigate directly to it.

[Go to resource](#)

## Create a private endpoint

Next, you'll need to create a private endpoint and configure it for disk access.

1. From your disk access resource, under **Settings**, select **Private endpoint connections**.

2. Select **+ Private endpoint**.

The screenshot shows the Azure portal interface for managing private endpoint connections for a disk access resource named 'example-disk-access-name'. The top navigation bar includes 'Disk Access' and 'Private endpoint connections'. The left sidebar lists 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Settings', 'Associated resources', 'Private endpoint connections' (which is highlighted with a red box), and 'Properties'. The main content area features a search bar, filter options for 'Filter by name...' and 'All connection states', and a table with columns for 'Connection name' and 'Connection state'. A message at the bottom states 'No results'.

3. In the **Create a private endpoint** pane, select a resource group.

- Provide a name and select the same region in which your disk access resource was created.

## Create a private endpoint

**1 Basics**   **2 Resource**   **3 Configuration**   **4 Tags**   **5 Review + create**

Use private endpoints to privately connect to a service or resource. Your private endpoint must be in the same region as your virtual network, but can be in a different region from the private link resource that you are connecting to. [Learn more](#)

### Project details

Subscription \* ⓘ

My Example Subscription

Resource group \* ⓘ

myResourceGroup

[Create new](#)

### Instance details

Name \* ⓘ

my-private-endpoint

Region \* ⓘ

(US) West US 2

- Select Next: Resource.

- On the Resource pane, select Connect to an Azure resource in my directory.

- For Resource type, select Microsoft.Compute/diskAccesses.

- For Resource, select the disk access resource you created earlier.

- Leave the Target sub-resource as disks.

## Create a private endpoint

**✓ Basics**   **✓ Resource**   **3 Configuration**   **4 Tags**   **5 Review + create**

Private Link offers options to create private endpoints for different Azure resources, like your private link service, a SQL server, or an Azure storage account. Select which resource you would like to connect to using this private endpoint. [Learn more](#)

Connection method ⓘ

Connect to an Azure resource in my directory.

Connect to an Azure resource by resource ID or alias.

Subscription \* ⓘ

My Example Subscription

Resource type \* ⓘ

Microsoft.Compute/diskAccesses

Resource \* ⓘ

example-disk-access-name

Target sub-resource \* ⓘ

disks

- Select Next : Configuration.

- Select the virtual network to which you will limit disk import and export. This prevents the import and export of your disk to other virtual networks.

### NOTE

If you have a network security group enabled for the selected subnet, it will be disabled for private endpoints on this subnet only. Other resources on this subnet will retain network security group enforcement.

12. Select the appropriate subnet.

## Create a private endpoint

✓ Basics ✓ Resource ③ Configuration ④ Tags ⑤ Review + create

**Networking**

To deploy the private endpoint, select a virtual network subnet. [Learn more](#)

Virtual network \* ⓘ myexamplergroup-vnet

Subnet \* ⓘ default (10.0.0.0/24)

ⓘ If you have a network security group (NSG) enabled for the subnet above, it will be disabled for private endpoints on this subnet only. Other resources on the subnet will still have NSG enforcement.

**Private DNS integration**

To connect privately with your private endpoint, you need a DNS record. We recommend that you integrate your private endpoint with a private DNS zone. You can also utilize your own DNS servers or create DNS records using the host files on your virtual machines. [Learn more](#)

Integrate with private DNS zone  Yes  No

Configuration name	Subscription	Private DNS zones
privatelink-blob-core-...	My Example Subscription	(New) privatelink.blob.core.windows.net

13. Select **Review + create**.

## Enable private endpoint on your disk

1. Navigate to the disk you'd like to configure.
2. Under **Settings**, select **Networking**.
3. Select **Private endpoint (through disk access)** and select the disk access you created earlier.

my-example-vm\_OsDisk\_1\_f06bcba8d92481194dce820b41a7356 | Networking

Save Discard

Search (Ctrl+ /)

Overview Activity log Access control (IAM) Tags

Settings

Configuration

Encryption

Networking

Disk Export Properties Locks Export template

**Network connectivity**

You can import or export from your managed disk either publicly or privately, using a private endpoint. To support disks at scale, a disk access resource is created to manage the private endpoints.

**Connectivity method**

Public endpoint (all networks)  Private endpoint (through disk access)  Deny all

**Disk access**

You can associate your managed disk with a disk access resource, which allows you to protect your disk with a private link. The disk access will only allow import and export operations through private endpoints. [Learn more](#)

Disk access \* ⓘ example-disk-access-name

4. Select **Save**.

You've now configured a private link that you can use to import and export your managed disk.

## Next steps

- Upload a VHD to Azure or copy a managed disk to another region - [Azure CLI](#) or [Azure PowerShell module](#)
- Download a VHD - [Windows](#) or [Linux](#)
- [FAQ for private links and managed disks](#)
- [Export/Copy managed snapshots as VHD to a storage account in different region with PowerShell](#)

# Upload a VHD to Azure or copy a managed disk to another region - Azure PowerShell

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article explains how to either upload a VHD from your local machine to an Azure managed disk or copy a managed disk to another region, using the Azure PowerShell module. The process of uploading a managed disk, also known as direct upload, enables you to upload a VHD up to 32 TiB in size directly into a managed disk. Currently, direct upload is supported for standard HDD, standard SSD, and premium SSDs. It isn't supported for ultra disks, yet.

If you're providing a backup solution for IaaS VMs in Azure, you should use direct upload to restore customer backups to managed disks. When uploading a VHD from a source external to Azure, speeds depend on your local bandwidth. When uploading or copying from an Azure VM, your bandwidth would be the same as standard HDDs.

## Secure uploads with Azure AD (preview)

If you're using [Azure Active Directory \(Azure AD\)](#) to control resource access, you can now use it to restrict uploading of Azure managed disks. This feature is currently in preview. When a user attempts to upload a disk, Azure validates the identity of the requesting user in Azure AD, and confirms that user has the required permissions. At a higher level, a system administrator could set a policy at the Azure account or subscription level to ensure that an Azure AD identity has the necessary permissions for uploading before allowing a disk or a disk snapshot to be uploaded. If you have any questions on securing uploads with Azure AD, reach out to this email: [azuredisks@microsoft.com](mailto:azuredisks@microsoft.com)

### Prerequisites

- Install the latest [Azure PowerShell module](#).
- You must enable the preview on your subscription, use the following command to enable the preview:

```
Register-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

It may take some time for the feature registration to complete, you can confirm if it has with the following command:

```
Get-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

### Restrictions

- VHDs can't be uploaded to empty snapshots.

### Assign RBAC role

To access managed disks secured with Azure AD, the requesting user must have either the [Data Operator for Managed Disks](#) role, or a [custom role](#) with the following permissions:

- Microsoft.Compute/disks/download/action
- Microsoft.Compute/disks/upload/action

- Microsoft.Compute/snapshots/download/action
- Microsoft.Compute/snapshots/upload/action

For detailed steps on assigning a role, see [Assign Azure roles using Azure PowerShell](#). To create or update a custom role, see [Create or update Azure custom roles using Azure PowerShell](#).

## Get started

There are two ways you can upload a VHD with the Azure PowerShell module: You can either use the [Add-AzVHD](#) command, which will automate most of the process for you, or you can perform the upload manually with AzCopy.

Generally, you should use [Add-AzVHD](#). However, if you need to upload a VHD that is larger than 50 GiB, consider [uploading the VHD manually with AzCopy](#). VHDs 50 GiB and larger upload faster using AzCopy.

For guidance on how to copy a managed disk from one region to another, see [Copy a managed disk](#).

## Use Add-AzVHD

### Prerequisites

- [Install the Azure PowerShell module](#).
- A VHD [has been prepared for Azure](#), stored locally.
  - On Windows: You don't need to convert your VHD to VHDx, convert it a fixed size, or resize it to include the 512-byte offset. `Add-AzVhd` performs these functions for you.
  - [Hyper-V](#) must be enabled for Add-AzVHD to perform these functions.
- On Linux: You must perform these actions manually. See [Resizing VHDs](#) for details.

### Upload a VHD

#### (Optional) Grant access to the disk

If Azure AD is used to enforce upload restrictions on a subscription or at the account level, [Add-AzVHD](#) only succeeds if attempted by a user that has the [appropriate RBAC role or necessary permissions](#). You'll need to [assign RBAC permissions](#) to grant access to the disk and generate a writeable SAS.

```
New-AzRoleAssignment -SignInName <emailOrUserprincipalname> `  
-RoleDefinitionName "Data Operator for Managed Disks" `  
-Scope /subscriptions/<subscriptionId>
```

### Use Add-AzVHD

The following example uploads a VHD from your local machine to a new Azure managed disk using [Add-AzVHD](#). Replace `<your-filepath-here>`, `<your-resource-group-name>`, `<desired-region>`, and `<desired-managed-disk-name>` with your parameters:

#### NOTE

If you're using Azure AD to enforce upload restrictions, add `DataAccessAuthMode 'AzureActiveDirectory'` to the end of your `Add-AzVhd` command.

```

# Required parameters
$path = <your-filepath-here>.vhf
$resourceGroup = <your-resource-group-name>
$location = <desired-region>
$name = <desired-managed-disk-name>

# Optional parameters
# $Zone = <desired-zone>
# $sku=<desired-SKU>
# -DataAccessAuthMode 'AzureActiveDirectory'

# To use $Zone or #sku, add -Zone or -DiskSKU parameters to the command
Add-AzVhd -LocalFilePath $path -ResourceGroupName $resourceGroup -Location $location -DiskName $name

```

## Manual upload

### Prerequisites

- Download the latest [version of AzCopy v10](#).
- [Install the Azure PowerShell module](#).
- A fixed size VHD that [has been prepared for Azure](#), stored locally.

To upload your VHD to Azure, you'll need to create an empty managed disk that is configured for this upload process. Before you create one, there's some additional information you should know about these disks.

This kind of managed disk has two unique states:

- ReadyToUpload, which means the disk is ready to receive an upload but, no [secure access signature](#) (SAS) has been generated.
- ActiveUpload, which means that the disk is ready to receive an upload and the SAS has been generated.

#### NOTE

While in either of these states, the managed disk will be billed at [standard HDD pricing](#), regardless of the actual type of disk. For example, a P10 will be billed as an S10. This will be true until `revoke-access` is called on the managed disk, which is required in order to attach the disk to a VM.

### Create an empty managed disk

Before you can create an empty standard HDD for uploading, you'll need the file size of the VHD you want to upload, in bytes. The example code will get that for you but, to do it yourself you can use:

`$vhdSizeBytes = (Get-Item "<fullFilePathHere>").length`. This value is used when specifying the -

**UploadSizeInBytes** parameter.

Now, on your local shell, create an empty standard HDD for uploading by specifying the **Upload** setting in the -**CreateOption** parameter as well as the **-UploadSizeInBytes** parameter in the [New-AzDiskConfig](#) cmdlet. Then call [New-AzDisk](#) to create the disk.

Replace `<yourdiskname>`, `<yourresourcegroupname>`, and `<yourregion>` then run the following commands:

#### TIP

If you're creating an OS disk, add `-HyperVGeneration '<yourGeneration>'` to `New-AzDiskConfig`.

If you're using Azure AD to secure your uploads, add `-DataAccessAuthMode 'AzureActiveDirectory'` to `New-AzDiskConfig`.

```

$vhdSizeBytes = (Get-Item "<fullFilePathHere>").length

$diskconfig = New-AzDiskConfig -SkuName 'Standard_LRS' -OsType 'Windows' -UploadSizeInBytes $vhdSizeBytes -
Location '<yourregion>' -CreateOption 'Upload'

New-AzDisk -ResourceGroupName '<yourresourcegroupname>' -DiskName '<yourdiskname>' -Disk $diskconfig

```

If you would like to upload either a premium SSD or a standard SSD, replace **Standard\_LRS** with either **Premium\_LRS** or **StandardSSD\_LRS**. Ultra disks aren't currently supported.

### Generate writeable SAS

Now that you've created an empty managed disk that is configured for the upload process, you can upload a VHD to it. To upload a VHD to the disk, you'll need a writeable SAS, so that you can reference it as the destination for your upload.

To generate a writable SAS of your empty managed disk, replace `<yourdiskname>` and `<yourresourcegroupname>`, then use the following commands:

```

$diskSas = Grant-AzDiskAccess -ResourceGroupName '<yourresourcegroupname>' -DiskName '<yourdiskname>' -
DurationInSecond 86400 -Access 'Write'

$disk = Get-AzDisk -ResourceGroupName '<yourresourcegroupname>' -DiskName '<yourdiskname>'

```

### Upload a VHD

Now that you have a SAS for your empty managed disk, you can use it to set your managed disk as the destination for your upload command.

Use AzCopy v10 to upload your local VHD file to a managed disk by specifying the SAS URI you generated.

This upload has the same throughput as the equivalent [standard HDD](#). For example, if you have a size that equates to S4, you will have a throughput of up to 60 MiB/s. But, if you have a size that equates to S70, you will have a throughput of up to 500 MiB/s.

```
AzCopy.exe copy "c:\somewhere\mydisk.vhd" $diskSas.AccessSAS --blob-type PageBlob
```

After the upload is complete, and you no longer need to write any more data to the disk, revoke the SAS. Revoking the SAS will change the state of the managed disk and allow you to attach the disk to a VM.

Replace `<yourdiskname>` and `<yourresourcegroupname>`, then run the following command:

```
Revoke-AzDiskAccess -ResourceGroupName '<yourresourcegroupname>' -DiskName '<yourdiskname>'
```

### Copy a managed disk

Direct upload also simplifies the process of copying a managed disk. You can either copy within the same region or copy your managed disk to another region.

The following script will do this for you, the process is similar to the steps described earlier, with some differences, since you're working with an existing disk.

## IMPORTANT

You must add an offset of 512 when you're providing the disk size in bytes of a managed disk from Azure. This is because Azure omits the footer when returning the disk size. The copy will fail if you don't do this. The following script already does this for you.

Replace the `<sourceResourceGroupHere>`, `<sourceDiskNameHere>`, `<targetDiskNameHere>`, `<targetResourceGroupHere>`, `<yourOSTypeHere>` and `<yourTargetLocationHere>` (an example of a location value would be uswest2) with your values, then run the following script in order to copy a managed disk.

## TIP

If you are creating an OS disk, add `-HyperVGeneration '<yourGeneration>'` to `New-AzDiskConfig`.

```
$sourceRG = <sourceResourceGroupHere>
$sourceDiskName = <sourceDiskNameHere>
$targetDiskName = <targetDiskNameHere>
$targetRG = <targetResourceGroupHere>
$targetLocate = <yourTargetLocationHere>
#Expected value for OS is either "Windows" or "Linux"
$targetOS = <yourOSTypeHere>

$sourceDisk = Get-AzDisk -ResourceGroupName $sourceRG -DiskName $sourceDiskName

# Adding the sizeInBytes with the 512 offset, and the -Upload flag
$targetDiskconfig = New-AzDiskConfig -SkuName 'Standard_LRS' -osType $targetOS -UploadSizeInBytes
$(($sourceDisk.DiskSizeBytes+512)) -Location $targetLocate -CreateOption 'Upload'

$targetDisk = New-AzDisk -ResourceGroupName $targetRG -DiskName $targetDiskName -Disk $targetDiskconfig

$sourceDiskSas = Grant-AzDiskAccess -ResourceGroupName $sourceRG -DiskName $sourceDiskName -DurationInSecond
86400 -Access 'Read'

$targetDiskSas = Grant-AzDiskAccess -ResourceGroupName $targetRG -DiskName $targetDiskName -DurationInSecond
86400 -Access 'Write'

azcopy copy $sourceDiskSas.AccessSAS $targetDiskSas.AccessSAS --blob-type PageBlob

Revoke-AzDiskAccess -ResourceGroupName $sourceRG -DiskName $sourceDiskName

Revoke-AzDiskAccess -ResourceGroupName $targetRG -DiskName $targetDiskName
```

## Next steps

Now that you've successfully uploaded a VHD to a managed disk, you can attach your disk to a VM and begin using it.

To learn how to attach a data disk to a VM, see our article on the subject: [Attach a data disk to a Windows VM with PowerShell](#). To use the disk as the OS disk, see [Create a Windows VM from a specialized disk](#).

# Upload a VHD to Azure or copy a managed disk to another region - Azure CLI

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article explains how to either upload a VHD from your local machine to an Azure managed disk or copy a managed disk to another region, using AzCopy. This process, direct upload, enables you to upload a VHD up to 32 TiB in size directly into a managed disk. Currently, direct upload is supported for standard HDD, standard SSD, and premium SSD managed disks. It isn't supported for ultra disks, yet.

If you're providing a backup solution for IaaS VMs in Azure, you should use direct upload to restore customer backups to managed disks. When uploading a VHD from a source external to Azure, speeds depend on your local bandwidth. When uploading or copying from an Azure VM, your bandwidth would be the same as standard HDDs.

## Secure uploads with Azure AD (preview)

If you're using [Azure Active Directory \(Azure AD\)](#) to control resource access, you can now use it to restrict uploading of Azure managed disks. This feature is currently in preview. When a user attempts to upload a disk, Azure validates the identity of the requesting user in Azure AD, and confirms that user has the required permissions. At a higher level, a system administrator could set a policy at the Azure account or subscription level, to ensure that an Azure AD identity has the necessary permissions for uploading before allowing a disk or a disk snapshot to be uploaded. If you have any questions on securing uploads with Azure AD, reach out to this email: [azuredisks@microsoft.com](mailto:azuredisks@microsoft.com)

### Prerequisites

- [Install the Azure CLI](#).
- Use the following command to enable the preview on your subscription:

```
az feature register --name AllowAADAuthForDataAccess --namespace Microsoft.Compute
```

It may take some time for the feature registration to complete, you can confirm if it has with the following command:

```
az feature show --name AllowAADAuthForDataAccess --namespace Microsoft.Compute --output table
```

### Restrictions

- VHDs can't be uploaded to empty snapshots.

### Assign RBAC role

To access managed disks secured with Azure AD, the requesting user must have either the [Data Operator for Managed Disks](#) role, or a [custom role](#) with the following permissions:

- `Microsoft.Compute/disks/download/action`
- `Microsoft.Compute/disks/upload/action`
- `Microsoft.Compute/snapshots/download/action`
- `Microsoft.Compute/snapshots/upload/action`

For detailed steps on assigning a role, see [Assign Azure roles using Azure CLI](#). To create or update a custom role, see [Create or update Azure custom roles using Azure CLI](#).

## Get started

If you'd prefer to upload disks through a GUI, you can do so using Azure Storage Explorer. For details refer to: [Use Azure Storage Explorer to manage Azure managed disks](#)

### Prerequisites

- Download the latest [version of AzCopy v10](#).
- [Install the Azure CLI](#).
- If you intend to upload a VHD from on-premises: A fixed size VHD that [has been prepared for Azure](#), stored locally.
- Or, a managed disk in Azure, if you intend to perform a copy action.

To upload your VHD to Azure, you'll need to create an empty managed disk that is configured for this upload process. Before you create one, there's some additional information you should know about these disks.

This kind of managed disk has two unique states:

- ReadToUpload, which means the disk is ready to receive an upload but, no [secure access signature](#) (SAS) has been generated.
- ActiveUpload, which means that the disk is ready to receive an upload and the SAS has been generated.

#### NOTE

While in either of these states, the managed disk will be billed at [standard HDD pricing](#), regardless of the actual type of disk. For example, a P10 will be billed as an S10. This will be true until `revoke-access` is called on the managed disk, which is required in order to attach the disk to a VM.

## Create an empty managed disk

Before you can create an empty standard HDD for uploading, you'll need the file size of the VHD you want to upload, in bytes. To get that, you can use either `wc -c <yourFileName>.vhf` or `ls -al <yourFileName>.vhf`. This value is used when specifying the `--upload-size-bytes` parameter.

Create an empty standard HDD for uploading by specifying both the `--for-upload` parameter and the `--upload-size-bytes` parameter in a [disk create cmdlet](#):

Replace `<yourdiskname>`, `<yourresourcegroupname>`, `<yourregion>` with values of your choosing. The `--upload-size-bytes` parameter contains an example value of `34359738880`, replace it with a value appropriate for you.

#### TIP

If you're creating an OS disk, add `--hyper-v-generation <yourGeneration>` to `az disk create`.

If you're using Azure AD to secure disk uploads, add `-dataAccessAuthmode 'AzureActiveDirectory'`.

```
az disk create -n <yourdiskname> -g <yourresourcegroupname> -l <yourregion> --os-type Linux --for-upload --upload-size-bytes 34359738880 --sku standard_lrs
```

If you would like to upload either a premium SSD or a standard SSD, replace `standard_lrs` with either

`premium_LRS` or `standardssd_lrs`. Ultra disks are not supported for now.

### (Optional) Grant access to the disk

If you're using Azure AD to secure uploads, you'll need to [assign RBAC permissions](#) to grant access to the disk and generate a writeable SAS.

```
az role assignment create --assignee "{assignee}" \
--role "{Data Operator for Managed Disks}" \
--scope
"/subscriptions/{subscriptionId}/resourcegroups/{resourceGroupName}/providers/{providerName}/{resourceType}/
{resourceSubType}/{diskName}"
```

### Generate writeable SAS

Now that you've created an empty managed disk that is configured for the upload process, you can upload a VHD to it. To upload a VHD to the disk, you'll need a writeable SAS, so that you can reference it as the destination for your upload.

To generate a writable SAS of your empty managed disk, replace `<yourdiskname>` and `<yourresourcegroupname>`, then use the following command:

```
az disk grant-access -n <yourdiskname> -g <yourresourcegroupname> --access-level Write --duration-in-seconds
86400
```

Sample returned value:

```
{
  "accessSas": "https://md-impexp-t0rdsfgsdfg4.blob.core.windows.net/w2c3mj0ksfg1/abcd?sv=2017-04-
17&sr=b&si=600a9281-d39e-4cc3-91d2-923c4a696537&sig=xTaT6mFgf139ycT87CADyFxb%2BnPXBElYirYRlbnJZbs%3D"
}
```

## Upload a VHD

Now that you have a SAS for your empty managed disk, you can use it to set your managed disk as the destination for your upload command.

Use AzCopy v10 to upload your local VHD file to a managed disk by specifying the SAS URI you generated.

This upload has the same throughput as the equivalent [standard HDD](#). For example, if you have a size that equates to S4, you will have a throughput of up to 60 MiB/s. But, if you have a size that equates to S70, you will have a throughput of up to 500 MiB/s.

```
AzCopy.exe copy "c:\somewhere\mydisk.vhd""sas-URI" --blob-type PageBlob
```

After the upload is complete, and you no longer need to write any more data to the disk, revoke the SAS. Revoking the SAS will change the state of the managed disk and allow you to attach the disk to a VM.

Replace `<yourdiskname>` and `<yourresourcegroupname>`, then use the following command to make the disk usable:

```
az disk revoke-access -n <yourdiskname> -g <yourresourcegroupname>
```

## Copy a managed disk

Direct upload also simplifies the process of copying a managed disk. You can either copy within the same region

or cross-region (to another region).

The following script will do this for you, the process is similar to the steps described earlier, with some differences since you're working with an existing disk.

#### IMPORTANT

You need to add an offset of 512 when you're providing the disk size in bytes of a managed disk from Azure. This is because Azure omits the footer when returning the disk size. The copy will fail if you don't do this. The following script already does this for you.

Replace the `<sourceResourceGroupHere>`, `<sourceDiskNameHere>`, `<targetDiskNameHere>`, `<targetResourceGroupHere>`, and `<yourTargetLocationHere>` (an example of a location value would be `uswest2`) with your values, then run the following script in order to copy a managed disk.

#### TIP

If you are creating an OS disk, add `--hyper-v-generation <yourGeneration>` to `az disk create`.

```
sourceDiskName=<sourceDiskNameHere>
sourceRG=<sourceResourceGroupHere>
targetDiskName=<targetDiskNameHere>
targetRG=<targetResourceGroupHere>
targetLocation=<yourTargetLocationHere>
#Expected value for OS is either "Windows" or "Linux"
targetOS=<yourOSTypeHere>

sourceDiskSizeBytes=$(az disk show -g $sourceRG -n $sourceDiskName --query '[diskSizeBytes]' -o tsv)

az disk create -g $targetRG -n $targetDiskName -l $targetLocation --os-type $targetOS --for-upload --upload-size-bytes $($sourceDiskSizeBytes+512) --sku standard_lrs

targetSASURI=$(az disk grant-access -n $targetDiskName -g $targetRG --access-level Write --duration-in-seconds 86400 -o tsv)

sourceSASURI=$(az disk grant-access -n $sourceDiskName -g $sourceRG --duration-in-seconds 86400 --query [accessSas] -o tsv)

azcopy copy $sourceSASURI $targetSASURI --blob-type PageBlob

az disk revoke-access -n $sourceDiskName -g $sourceRG

az disk revoke-access -n $targetDiskName -g $targetRG
```

## Next steps

Now that you've successfully uploaded a VHD to a managed disk, you can attach the disk as a [data disk to an existing VM](#) or [attach the disk to a VM as an OS disk](#), to create a new VM.

# Download a Linux VHD from Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In this article, you learn how to download a Linux virtual hard disk (VHD) file from Azure using the Azure portal.

## Stop the VM

A VHD can't be downloaded from Azure if it's attached to a running VM. If you want to keep the VM running, you can [create a snapshot and then download the snapshot](#).

To stop the VM:

1. Sign in to the [Azure portal](#).
2. On the left menu, select **Virtual Machines**.
3. Select the VM from the list.
4. On the page for the VM, select **Stop**.



### Alternative: Snapshot the VM disk

Take a snapshot of the disk to download.

1. Select the VM in the [portal](#).
2. Select **Disk** in the left menu and then select the disk you want to snapshot. The details of the disk will be displayed.
3. Select **Create Snapshot** from the menu at the top of the page. The **Create snapshot** page will open.
4. In **Name**, type a name for the snapshot.
5. For **Snapshot type**, select **Full** or **Incremental**.
6. When you are done, select **Review + create**.

Your snapshot will be created shortly, and can then be used to download or create another VM.

#### NOTE

If you don't stop the VM first, the snapshot will not be clean. The snapshot will be in the same state as if the VM had been power cycled or crashed at the point in time when the snapshot was made. While usually safe, it could cause problems if the running applications running at the time were not crash resistant.

This method is only recommended for VMs with a single OS disk. VMs with one or more data disks should be stopped before download or before creating a snapshot for the OS disk and each data disk.

## Secure downloads and uploads with Azure AD (preview)

If you're using [Azure Active Directory \(Azure AD\)](#) to control resource access, you can now use it to restrict uploads and downloads of Azure managed disks. This feature is currently in preview. When a user attempts to upload or download a disk, Azure validates the identity of the requesting user in Azure AD, and confirms that user has the required permissions. At a higher level, a system administrator could set a policy at the Azure

account or subscription level, to ensure that all disks and snapshots must use Azure AD for uploads or downloads. If you have any questions on securing uploads or downloads with Azure AD, reach out to this email: [azuredisks@microsoft.com](mailto:azuredisks@microsoft.com)

## Restrictions

- VHDs can't be uploaded to empty snapshots.
- To download a VHD that is using Azure AD to restrict access, you must access the Azure portal from this link: <https://aka.ms/dataAccessAuthenticationMode>

## Prerequisites

- Install the latest [Azure PowerShell module](#).
- You must enable the preview on your subscription, use the following command to enable the preview:

```
Register-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

It may take some time for the feature registration to complete, you can confirm if it has with the following command:

```
Get-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

## Assign RBAC role

To access managed disks secured with Azure AD, the requesting user must have either the [Data Operator for Managed Disks](#) role, or a [custom role](#) with the following permissions:

- Microsoft.Compute/disks/download/action
- Microsoft.Compute/disks/upload/action
- Microsoft.Compute/snapshots/download/action
- Microsoft.Compute/snapshots/upload/action

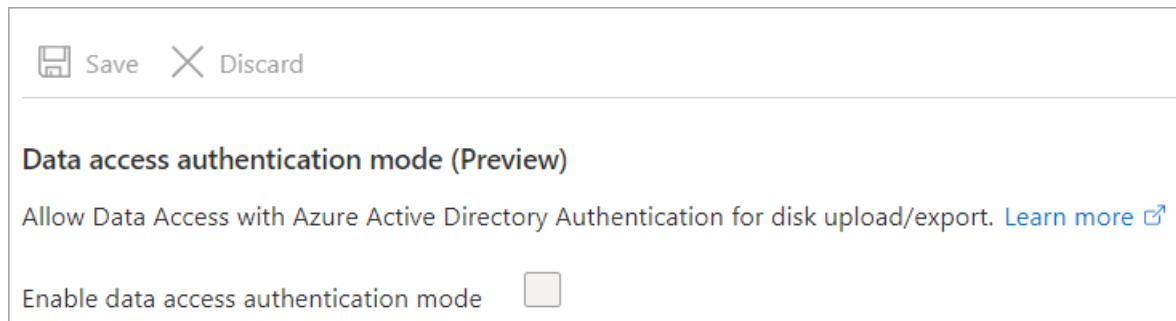
For detailed steps on assigning a role, see the following articles for [portal](#), [PowerShell](#), or [CLI](#). To create or update a custom role, see the following articles for [portal](#), [PowerShell](#), or [CLI](#).

## Enable data access authentication mode

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

Enable **data access authentication mode** to restrict access to the disk. You can either enable it when creating the disk, or you can enable it on the **Disk Export** page for existing disks. In order to enable **data access authentication mode** you must access the Azure portal from the following link:

<https://aka.ms/dataAccessAuthenticationMode>



## Generate SAS URL

To download the VHD file, you need to generate a [shared access signature \(SAS\)](#) URL. When the URL is generated, an expiration time is assigned to the URL.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. On the menu of the page for the VM, select **Disks**.
2. Select the operating system disk for the VM, and then select **Disk Export**.
3. If required, update the value of **URL expires in (seconds)** to give you enough time to complete the download. The default is 3600 seconds (one hour).
4. Select **Generate URL**.

## Download VHD

### NOTE

If you're using Azure AD to secure managed disk downloads, the user downloading the VHD must have the appropriate [RBAC permissions](#).

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Under the URL that was generated, select **Download the VHD file**.

The following URL can be used to download the VHD file for this disk. Copy it and keep it secure, it will not be shown again.

<https://md-sm1xb0fsrmql.blob.core.windows.net/1sv2pbtcgcqz/abcd?sv=2017-04-17&sr=b&si=1c73f2e3-09bc-45ac-a4c1-7d0bb2...> 

[Download the VHD file](#)

A SAS URL has been generated for this disk for export. While in this state, it can't be edited or attached to a running virtual machine. You can cancel the export by clicking the button below. This will revoke the SAS URL, and may cancel any in-progress transfers if the disk is currently being downloaded to another location.

[Cancel export](#)

2. You may need to select **Save** in the browser to start the download. The default name for the VHD file is *abcd*.

## Next steps

- Learn how to [upload and create a Linux VM from custom disk with the Azure CLI](#).
- [Manage Azure disks the Azure CLI](#).

# Download a Windows VHD from Azure

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

In this article, you learn how to download a Windows virtual hard disk (VHD) file from Azure using the Azure portal.

## Optional: Generalize the VM

If you want to use the VHD as an [image](#) to create other VMs, you should use [Sysprep](#) to generalize the operating system. Otherwise, you will have to make a copy of the disk for each VM you want to create.

To use the VHD as an image to create other VMs, generalize the VM.

1. If you haven't already done so, sign in to the [Azure portal](#).
2. [Connect to the VM](#).
3. On the VM, open the Command Prompt window as an administrator.
4. Change the directory to `%windir%\system32\sysprep` and run `sysprep.exe`.
5. In the System Preparation Tool dialog box, select **Enter System Out-of-Box Experience (OOBE)**, and make sure that **Generalize** is selected.
6. In Shutdown Options, select **Shutdown**, and then click **OK**.

If you don't want to generalize your current VM, you can still create a generalized image by first [making a snapshot of the OS disk](#), creating a new VM from the snapshot, and then generalizing the copy.

## Stop the VM

A VHD can't be downloaded from Azure if it's attached to a running VM. If you want to keep the VM running, you can [create a snapshot and then download the snapshot](#).

1. On the Hub menu in the Azure portal, click **Virtual Machines**.
2. Select the VM from the list.
3. On the blade for the VM, click **Stop**.

## Alternative: Snapshot the VM disk

Take a snapshot of the disk to download.

1. Select the VM in the [portal](#).
2. Select **Disk** in the left menu and then select the disk you want to snapshot. The details of the disk will be displayed.
3. Select **Create Snapshot** from the menu at the top of the page. The **Create snapshot** page will open.
4. In **Name**, type a name for the snapshot.
5. For **Snapshot type**, select **Full** or **Incremental**.
6. When you are done, select **Review + create**.

Your snapshot will be created shortly, and can then be used to download or create another VM.

#### **NOTE**

If you don't stop the VM first, the snapshot will not be clean. The snapshot will be in the same state as if the VM had been power cycled or crashed at the point in time when the snapshot was made. While usually safe, it could cause problems if the running applications running at the time were not crash resistant.

This method is only recommended for VMs with a single OS disk. VMs with one or more data disks should be stopped before download or before creating a snapshot for the OS disk and each data disk.

## Secure downloads and uploads with Azure AD (preview)

If you're using [Azure Active Directory \(Azure AD\)](#) to control resource access, you can now use it to restrict uploads and downloads of Azure managed disks. This feature is currently in preview. When a user attempts to upload or download a disk, Azure validates the identity of the requesting user in Azure AD, and confirms that user has the required permissions. At a higher level, a system administrator could set a policy at the Azure account or subscription level, to ensure that all disks and snapshots must use Azure AD for uploads or downloads. If you have any questions on securing uploads or downloads with Azure AD, reach out to this email: [azuredisks@microsoft.com](mailto:azuredisks@microsoft.com)

#### **Restrictions**

- VHDS can't be uploaded to empty snapshots.
- To download a VHD that is using Azure AD to restrict access, you must access the Azure portal from this link: <https://aka.ms/dataAccessAuthenticationMode>

#### **Prerequisites**

- Install the latest [Azure PowerShell module](#).
- You must enable the preview on your subscription, use the following command to enable the preview:

```
Register-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

It may take some time for the feature registration to complete, you can confirm if it has with the following command:

```
Get-AzProviderFeature -FeatureName "AllowAADAuthForDataAccess" -ProviderNamespace "Microsoft.Compute"
```

#### **Assign RBAC role**

To access managed disks secured with Azure AD, the requesting user must have either the [Data Operator for Managed Disks](#) role, or a [custom role](#) with the following permissions:

- Microsoft.Compute/disks/download/action
- Microsoft.Compute/disks/upload/action
- Microsoft.Compute/snapshots/download/action
- Microsoft.Compute/snapshots/upload/action

For detailed steps on assigning a role, see the following articles for [portal](#), [PowerShell](#), or [CLI](#). To create or update a custom role, see the following articles for [portal](#), [PowerShell](#), or [CLI](#).

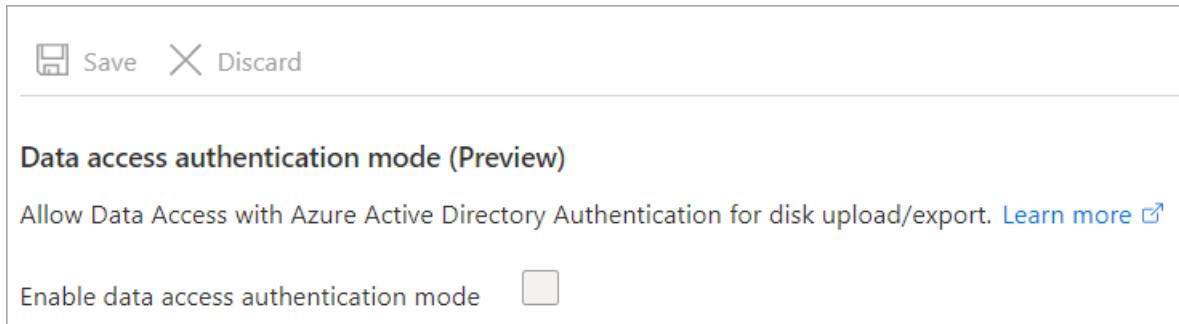
#### **Enable data access authentication mode**

- [Portal](#)
- [PowerShell](#)

- [Azure CLI](#)

Enable **data access authentication mode** to restrict access to the disk. You can either enable it when creating the disk, or you can enable it on the **Disk Export** page for existing disks. In order to enable **data access authentication mode** you must access the Azure portal from the following link:

<https://aka.ms/dataAccessAuthenticationMode>



## Generate download URL

To download the VHD file, you need to generate a [shared access signature \(SAS\)](#) URL. When the URL is generated, an expiration time is assigned to the URL.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. On the page for the VM, click **Disks** in the left menu.
2. Select the operating system disk for the VM.
3. On the page for the disk, select **Disk Export** from the left menu.
4. The default expiration time of the URL is 3600 seconds (one hour). You may need to increase this for Windows OS disks or large data disks. 36000 seconds (10 hours) is usually sufficient.
5. Click **Generate URL**.

### NOTE

The expiration time is increased from the default to provide enough time to download the large VHD file for a Windows Server operating system. Large VHDs can take up to several hours to download depending on your connection and the size of the VM.

## Download VHD

### NOTE

If you're using Azure AD to secure managed disk downloads, the user downloading the VHD must have the appropriate RBAC permissions.

- [Portal](#)
- [PowerShell](#)
- [Azure CLI](#)

1. Under the URL that was generated, click Download the VHD file.
2. You may need to click **Save** in your browser to start the download. The default name for the VHD file is *abcd.vhd*.

## Next steps

- Learn how to upload a VHD file to Azure.
- [Create managed disks from unmanaged disks in a storage account](#).
- [Manage Azure disks with PowerShell](#).

# Convert Azure managed disks storage from Standard to Premium or Premium to Standard

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

There are four disk types of Azure managed disks: Azure ultra disks, premium SSD, standard SSD, and standard HDD. You can switch between premium SSD, standard SSD, and standard HDD based on your performance needs. You are not yet able to switch from or to an ultra disk, you must deploy a new one.

This functionality is not supported for unmanaged disks. But you can easily [convert an unmanaged disk to a managed disk](#) to be able to switch between disk types.

This article shows how to convert managed disks from one disk type to another by using the Azure CLI. To install or upgrade the tool, see [Install Azure CLI](#).

## Before you begin

- Disk conversion requires a restart of the virtual machine (VM), so schedule the migration of your disk storage during a pre-existing maintenance window.
- For unmanaged disks, first [convert to managed disks](#) so you can switch between storage options.

## Switch all managed disks of a VM between from one account to another

This example shows how to convert all of a VM's disks to premium storage. However, by changing the sku variable in this example, you can convert the VM's disks type to standard SSD or standard HDD. Please note that To use Premium managed disks, your VM must use a [VM size](#) that supports Premium storage. This example also switches to a size that supports Premium storage.

```

#resource group that contains the virtual machine
rgName='yourResourceGroup'

#Name of the virtual machine
vmName='yourVM'

#Premium capable size
#Required only if converting from Standard to Premium
size='Standard_DS2_v2'

#Choose between Standard_LRS, StandardSSD_LRS and Premium_LRS based on your scenario
sku='Premium_LRS'

#Deallocate the VM before changing the size of the VM
az vm deallocate --name $vmName --resource-group $rgName

#Change the VM size to a size that supports Premium storage
#Skip this step if converting storage from Premium to Standard
az vm resize --resource-group $rgName --name $vmName --size $size

#Update the SKU of all the data disks
az vm show -n $vmName -g $rgName --query storageProfile.dataDisks[*].managedDisk -o tsv \
| awk -v sku=$sku '{system("az disk update --sku \"sku\" --ids \"$1\"")}' 

#Update the SKU of the OS disk
az vm show -n $vmName -g $rgName --query storageProfile.osDisk.managedDisk -o tsv \
| awk -v sku=$sku '{system("az disk update --sku \"sku\" --ids \"$1\"")}' 

az vm start --name $vmName --resource-group $rgName

```

## Switch individual managed disks from one disk type to another

For your dev/test workload, you might want to have a mix of Standard and Premium disks to reduce your costs. You can choose to upgrade only those disks that need better performance. This example shows how to convert a single VM disk from Standard to Premium storage. However, by changing the sku variable in this example, you can convert the VM's disks type to standard SSD or standard HDD. To use Premium managed disks, your VM must use a [VM size](#) that supports Premium storage. This example also switches to a size that supports Premium storage.

```

#resource group that contains the managed disk
rgName='yourResourceGroup'

#Name of your managed disk
diskName='yourManagedDiskName'

#Premium capable size
#Required only if converting from Standard to Premium
size='Standard_DS2_v2'

#Choose between Standard_LRS, StandardSSD_LRS and Premium_LRS based on your scenario
sku='Premium_LRS'

#Get the parent VM Id
vmId=$(az disk show --name $diskName --resource-group $rgName --query managedBy --output tsv)

#Deallocate the VM before changing the size of the VM
az vm deallocate --ids $vmId

#Change the VM size to a size that supports Premium storage
#Skip this step if converting storage from Premium to Standard
az vm resize --ids $vmId --size $size

# Update the SKU
az disk update --sku $sku --name $diskName --resource-group $rgName

az vm start --ids $vmId

```

## Switch managed disks from one disk type to another

Follow these steps:

1. Sign in to the [Azure portal](#).
2. Select the VM from the list of **Virtual machines**.
3. If the VM isn't stopped, select **Stop** at the top of the **VM Overview** pane, and wait for the VM to stop.
4. In the pane for the VM, select **Disk**s from the menu.
5. Select the disk that you want to convert.
6. Select **Configuration** from the menu.
7. Change the **Account type** from the original disk type to the desired disk type.
8. Select **Save**, and close the disk pane.

The update of the disk type is instantaneous. You can restart your VM after the conversion.

## Next steps

Make a read-only copy of a VM by using [snapshots](#).

# Update the storage type of a managed disk

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows

There are four disk types of Azure managed disks: Azure ultra disks, premium SSD, standard SSD, and standard HDD. You can switch between premium SSD, standard SSD, and standard HDD based on your performance needs. You are not yet able to switch from or to an ultra disk, you must deploy a new one.

This functionality is not supported for unmanaged disks. But you can easily [convert an unmanaged disk to a managed disk](#) to be able to switch between disk types.

## Before you begin

- Because conversion requires a restart of the virtual machine (VM), you should schedule the migration of your disk storage during a pre-existing maintenance window.
- If your disk is unmanaged, first [convert it to a managed disk](#) so you can switch between storage options.

## Switch all managed disks of a VM between from one account to another

This example shows how to convert all of a VM's disks to premium storage. However, by changing the \$storageType variable in this example, you can convert the VM's disks type to standard SSD or standard HDD. To use Premium managed disks, your VM must use a [VM size](#) that supports Premium storage. This example also switches to a size that supports premium storage:

```

# Name of the resource group that contains the VM
$rgName = 'yourResourceGroup'

# Name of your virtual machine
$vmName = 'yourVM'

# Choose between Standard_LRS, StandardSSD_LRS and Premium_LRS based on your scenario
$storageType = 'Premium_LRS'

# Premium capable size
# Required only if converting storage from Standard to Premium
$size = 'Standard_DS2_v2'

# Stop and deallocate the VM before changing the size
Stop-AzVM -ResourceGroupName $rgName -Name $vmName -Force

$vm = Get-AzVM -Name $vmName -resourceGroupName $rgName

# Change the VM size to a size that supports Premium storage
# Skip this step if converting storage from Premium to Standard
$vm.HardwareProfile.VmSize = $size
Update-AzVM -VM $vm -ResourceGroupName $rgName

# Get all disks in the resource group of the VM
$vmDisks = Get-AzDisk -ResourceGroupName $rgName

# For disks that belong to the selected VM, convert to Premium storage
foreach ($disk in $vmDisks)
{
    if ($disk.ManagedBy -eq $vm.Id)
    {
        $disk.Sku = [Microsoft.Azure.Management.Compute.Models.DiskSku]::new($storageType)
        $disk | Update-AzDisk
    }
}

Start-AzVM -ResourceGroupName $rgName -Name $vmName

```

## Switch individual managed disks between Standard and Premium

For your dev/test workload, you might want a mix of Standard and Premium disks to reduce your costs. You can choose to upgrade only those disks that need better performance. This example shows how to convert a single VM disk from Standard to Premium storage. However, by changing the \$storageType variable in this example, you can convert the VM's disks type to standard SSD or standard HDD. To use Premium managed disks, your VM must use a [VM size](#) that supports Premium storage. This example also shows how to switch to a size that supports Premium storage:

```

$diskName = 'yourDiskName'
# resource group that contains the managed disk
$rgName = 'yourResourceGroupName'
# Choose between Standard_LRS, StandardSSD_LRS and Premium_LRS based on your scenario
$storageType = 'Premium_LRS'
# Premium capable size
$size = 'Standard_DS2_v2'

$disk = Get-AzDisk -DiskName $diskName -ResourceGroupName $rgName

# Get parent VM resource
$vmResource = Get-AzResource -ResourceId $disk.ManagedBy

# Stop and deallocate the VM before changing the storage type
Stop-AzVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name -Force

$vm = Get-AzVM -ResourceGroupName $vmResource.ResourceGroupName -Name $vmResource.Name

# Change the VM size to a size that supports Premium storage
# Skip this step if converting storage from Premium to Standard
$vm.HardwareProfile.VmSize = $size
Update-AzVM -VM $vm -ResourceGroupName $rgName

# Update the storage type
$disk.Sku = [Microsoft.Azure.Management.Compute.Models.DiskSku]::new($storageType)
$disk | Update-AzDisk

Start-AzVM -ResourceGroupName $vm.ResourceGroupName -Name $vm.Name

```

## Switch managed disks from one disk type to another

Follow these steps:

1. Sign in to the [Azure portal](#).
2. Select the VM from the list of **Virtual machines**.
3. If the VM isn't stopped, select **Stop** at the top of the **VM Overview** pane, and wait for the VM to stop.
4. In the pane for the VM, select **Disks** from the menu.
5. Select the disk that you want to convert.
6. Select **Size + performance** from the menu.
7. Change the **Account type** from the original disk type to the desired disk type.
8. Select **Save**, and close the disk pane.

The disk type conversion is instantaneous. You can start your VM after the conversion.

## Next steps

Make a read-only copy of a VM by using a [snapshot](#).

# Use Site Recovery to migrate to Premium Storage

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

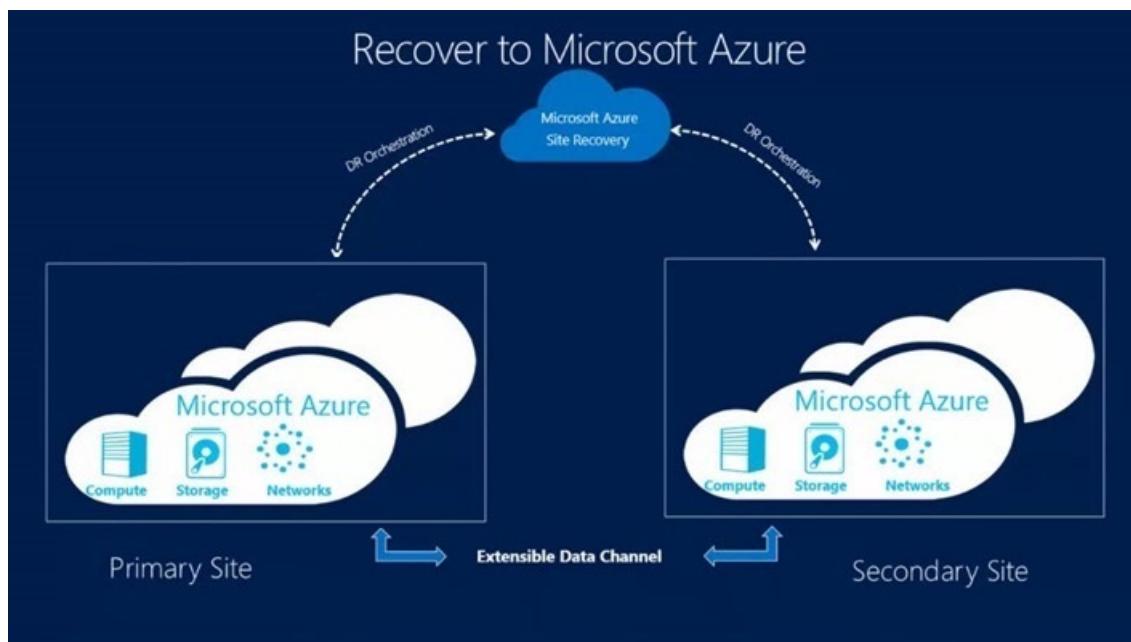
Azure premium SSDs delivers high-performance, low-latency disk support for virtual machines (VMs) that are running I/O-intensive workloads. This guide helps you migrate your VM disks from a standard storage account to a premium storage account by using [Azure Site Recovery](#).

Site Recovery is an Azure service that contributes to your strategy for business continuity and disaster recovery by orchestrating the replication of on-premises physical servers and VMs to the cloud (Azure) or to a secondary datacenter. When outages occur in your primary location, you fail over to the secondary location to keep applications and workloads available. You fail back to your primary location when it returns to normal operation.

Site Recovery provides test failovers to support disaster recovery drills without affecting production environments. You can run failovers with minimal data loss (depending on replication frequency) for unexpected disasters. In the scenario of migrating to Premium Storage, you can use the [failover in Site Recovery](#) to migrate target disks to a premium storage account.

We recommend migrating to Premium Storage by using Site Recovery because this option provides minimal downtime. This option also avoids the manual execution of copying disks and creating new VMs. Site Recovery will systematically copy your disks and create new VMs during failover.

Site Recovery supports a number of types of failover with minimal or no downtime. To plan your downtime and estimate data loss, see the [types of failover in Site Recovery](#). If you [prepare to connect to Azure VMs after failover](#), you should be able to connect to the Azure VM by using RDP after failover.



## Azure Site Recovery components

These Site Recovery components are relevant to this migration scenario:

- **Configuration server** is an Azure VM that coordinates communication and manages data replication and recovery processes. On this VM, you run a single setup file to install the configuration server and an additional component, called a process server, as a replication gateway. Read about [configuration server](#)

prerequisites. You set up the configuration server only once, and you can use it for all migrations to the same region.

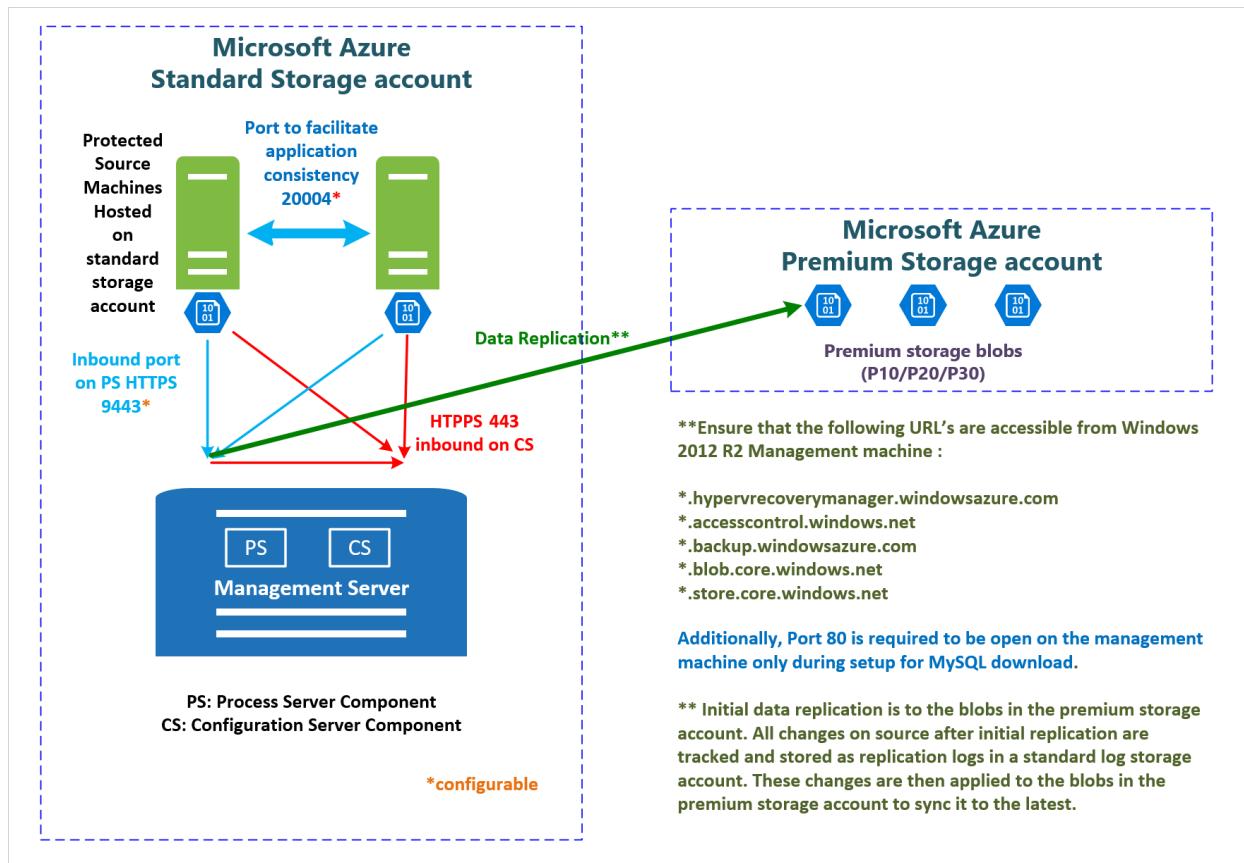
- **Process server** is a replication gateway that:

1. Receives replication data from source VMs.
2. Optimizes the data with caching, compression, and encryption.
3. Sends the data to a storage account.

It also handles push installation of the mobility service to source VMs and performs automatic discovery of source VMs. The default process server is installed on the configuration server. You can deploy additional standalone process servers to scale your deployment. Read about [best practices for process server deployment](#) and [deploying additional process servers](#). You set up the process server only once, and you can use it for all migrations to the same region.

- **Mobility service** is a component that is deployed on every standard VM that you want to replicate. It captures data writes on the standard VM and forwards them to the process server. Read about [replicated machine prerequisites](#).

This graphic shows how these components interact:



#### NOTE

Site Recovery does not support the migration of Storage Spaces disks.

For additional components for other scenarios, see [Scenario architecture](#).

## Azure essentials

These are the Azure requirements for this migration scenario:

- An Azure subscription.

- An Azure premium storage account to store replicated data.
- An Azure virtual network to which VMs will connect when they're created at failover. The Azure virtual network must be in the same region as the one in which Site Recovery runs.
- An Azure standard storage account to store replication logs. This can be the same storage account for the VM disks that are being migrated.

## Prerequisites

- Understand the relevant migration scenario components in the preceding section.
- Plan your downtime by learning about [failover in Site Recovery](#).

## Setup and migration steps

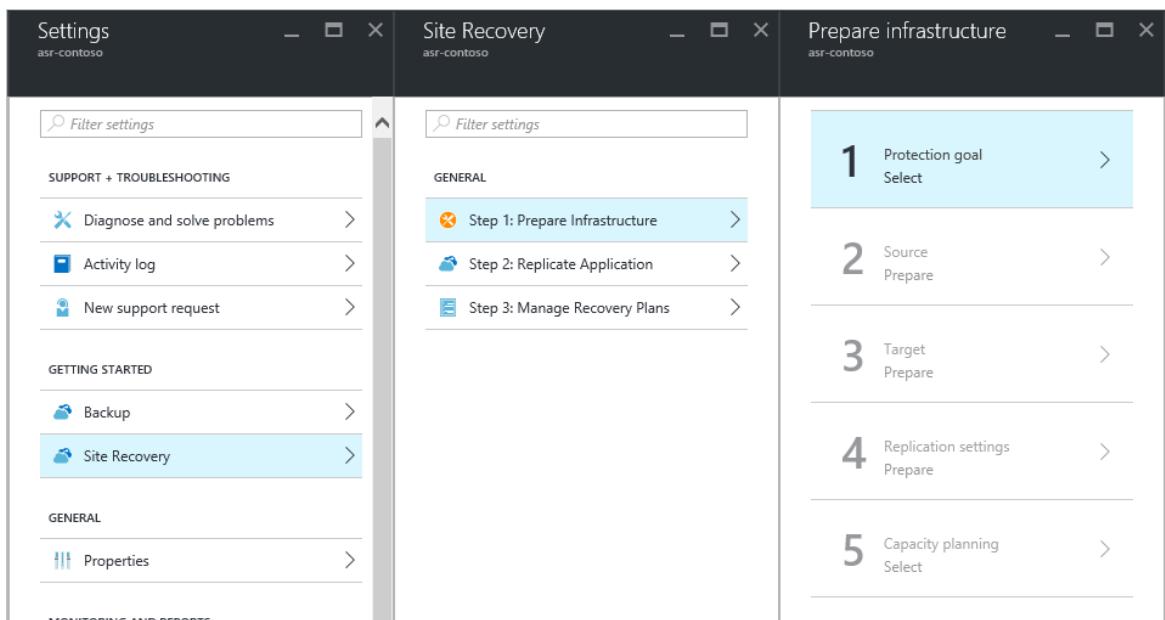
You can use Site Recovery to migrate Azure IaaS VMs between regions or within same region. The following instructions are tailored for this migration scenario from the article [Replicate VMware VMs or physical servers to Azure](#). Please follow the links for detailed steps in addition to the instructions in this article.

### Step 1: Create a Recovery Services vault

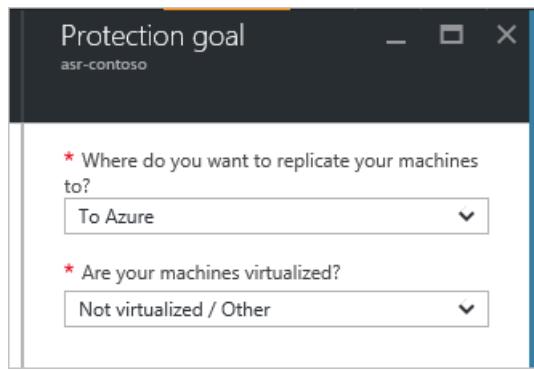
1. Open the [Azure portal](#).
2. Select **Create a resource > Management > Backup and Site Recovery (OMS)**. Alternatively, you can select **Browse > Recovery Services Vault > Add**.
3. Specify a region that VMs will be replicated to. For the purpose of migration in the same region, select the region where your source VMs and source storage accounts are.

### Step 2: Choose your protection goals

1. On the VM where you want to install the configuration server, open the [Azure portal](#).
2. Go to **Recovery Services vaults > Settings > Site Recovery > Step 1: Prepare Infrastructure > Protection goal**.



3. Under **Protection goal**, in the first drop-down list, select **To Azure**. In the second drop-down list, select **Not virtualized / Other**, and then select **OK**.

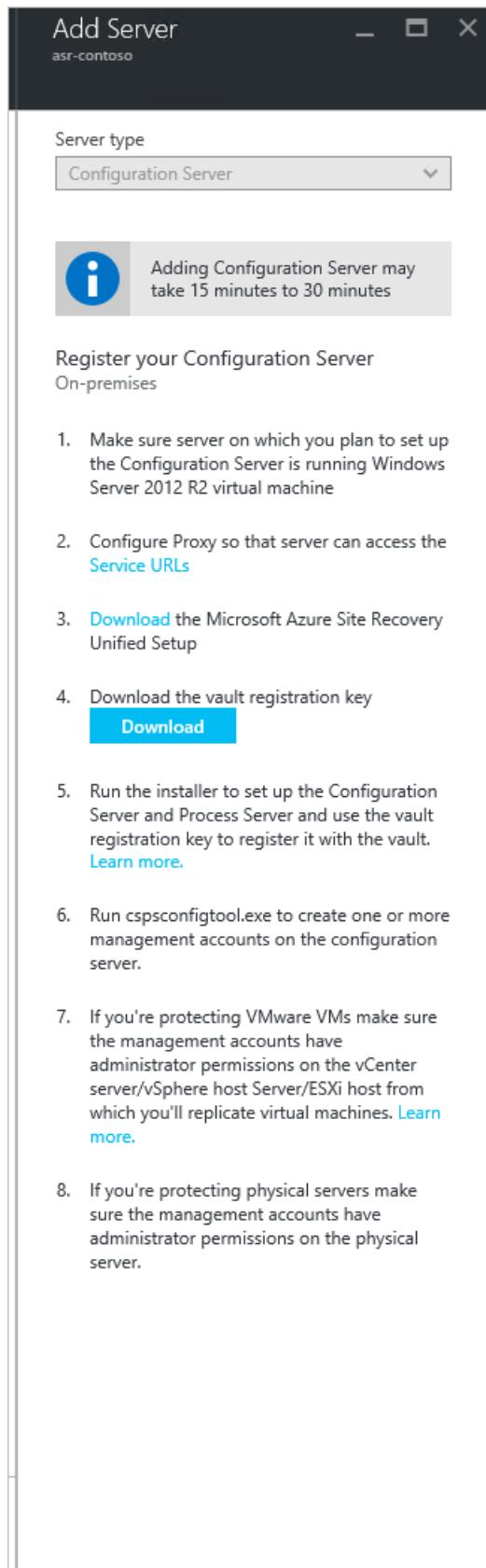


### Step 3: Set up the source environment (configuration server)

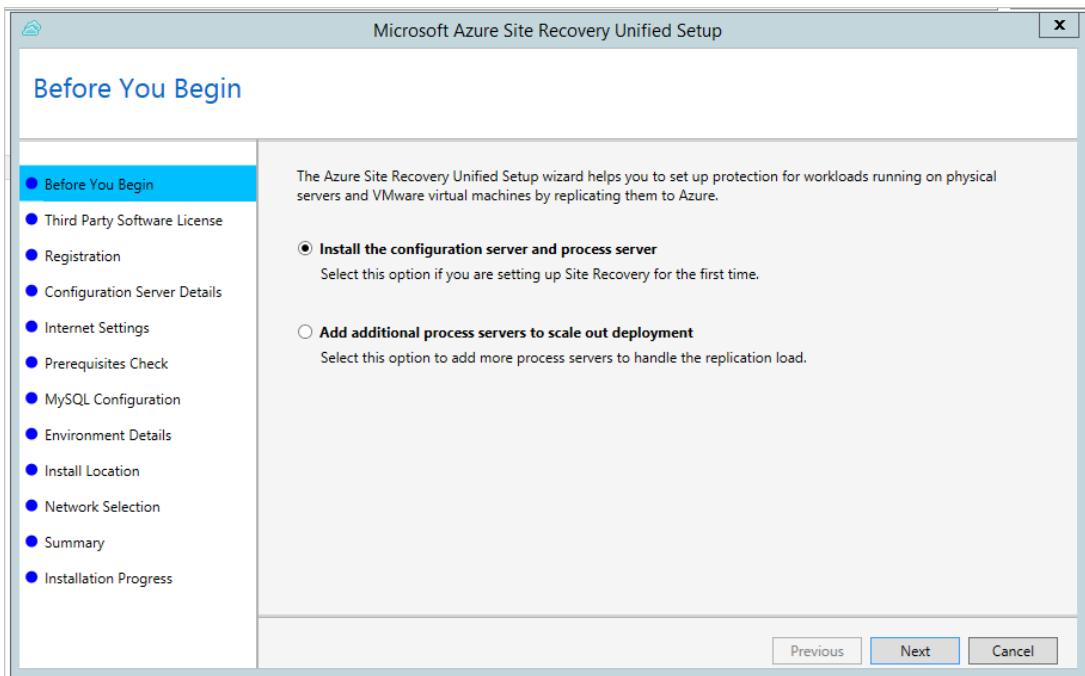
1. Download Azure Site Recovery Unified Setup and the vault registration key by going to the **Prepare infrastructure** > **Prepare source** > **Add Server** panes.

You will need the vault registration key to run the unified setup. The key is valid for five days after you generate it.

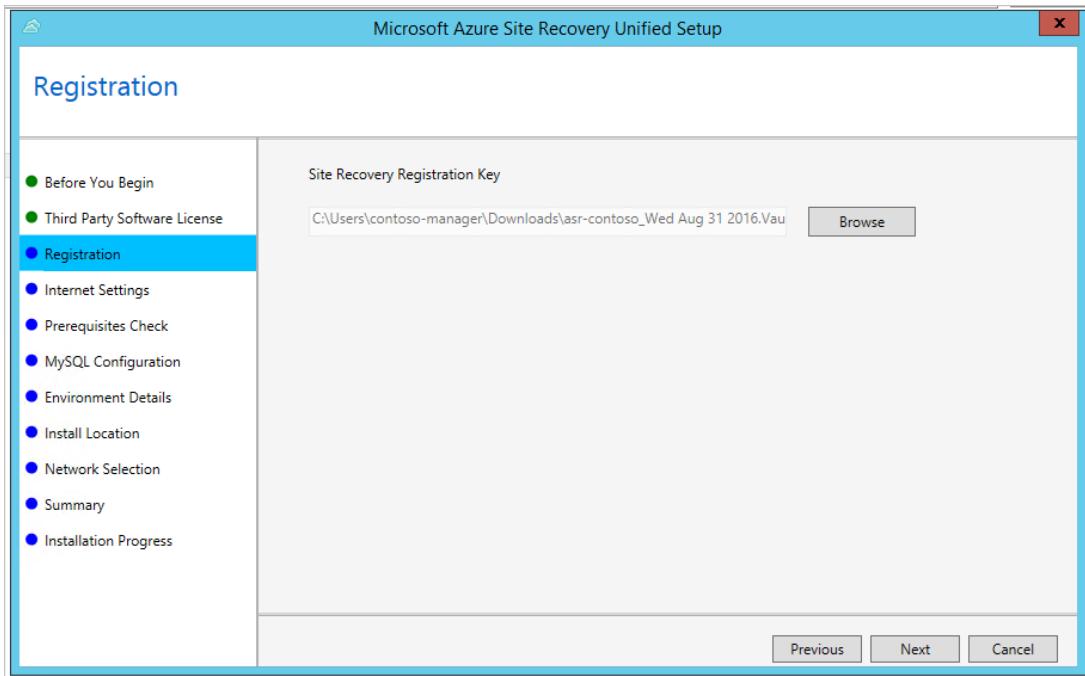
2. In the Add Server pane, add a configuration server.



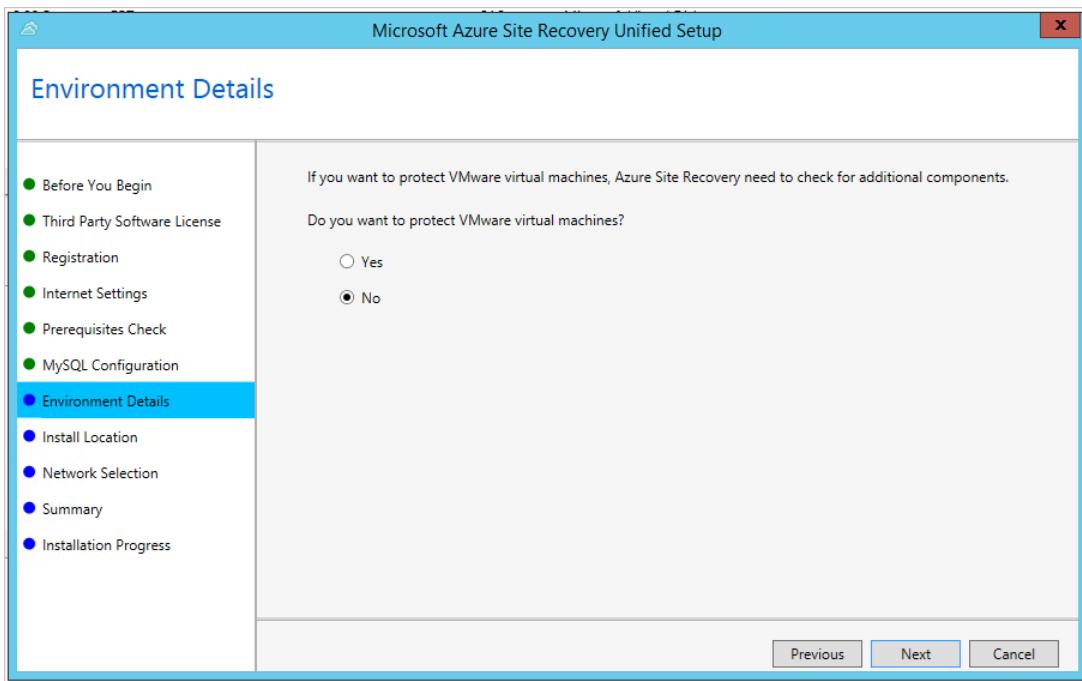
3. On the VM that you're using as the configuration server, run Unified Setup to install the configuration server and the process server. You can [walk through the screenshots](#) to complete the installation. You can refer to the following screenshots for steps specified for this migration scenario.
  - a. In **Before You Begin**, select **Install the configuration server and process server**.



- b. In **Registration**, browse and select the registration key that you downloaded from the vault.

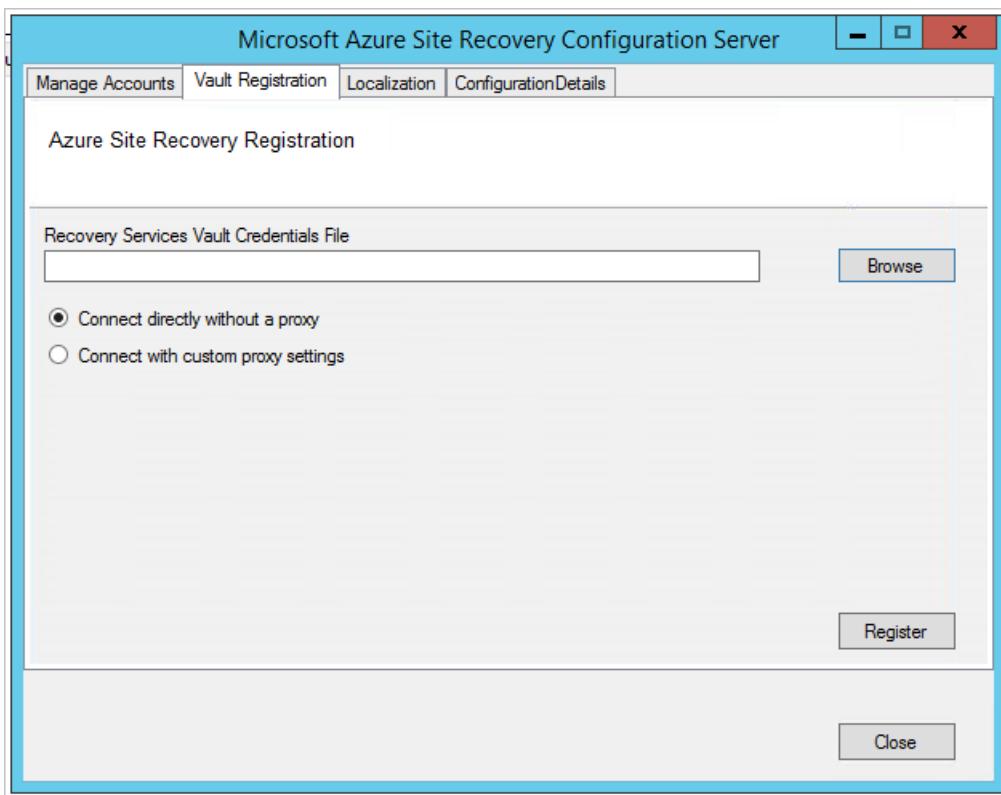


- c. In **Environment Details**, select whether you're going to replicate VMware VMs. For this migration scenario, choose **No**.



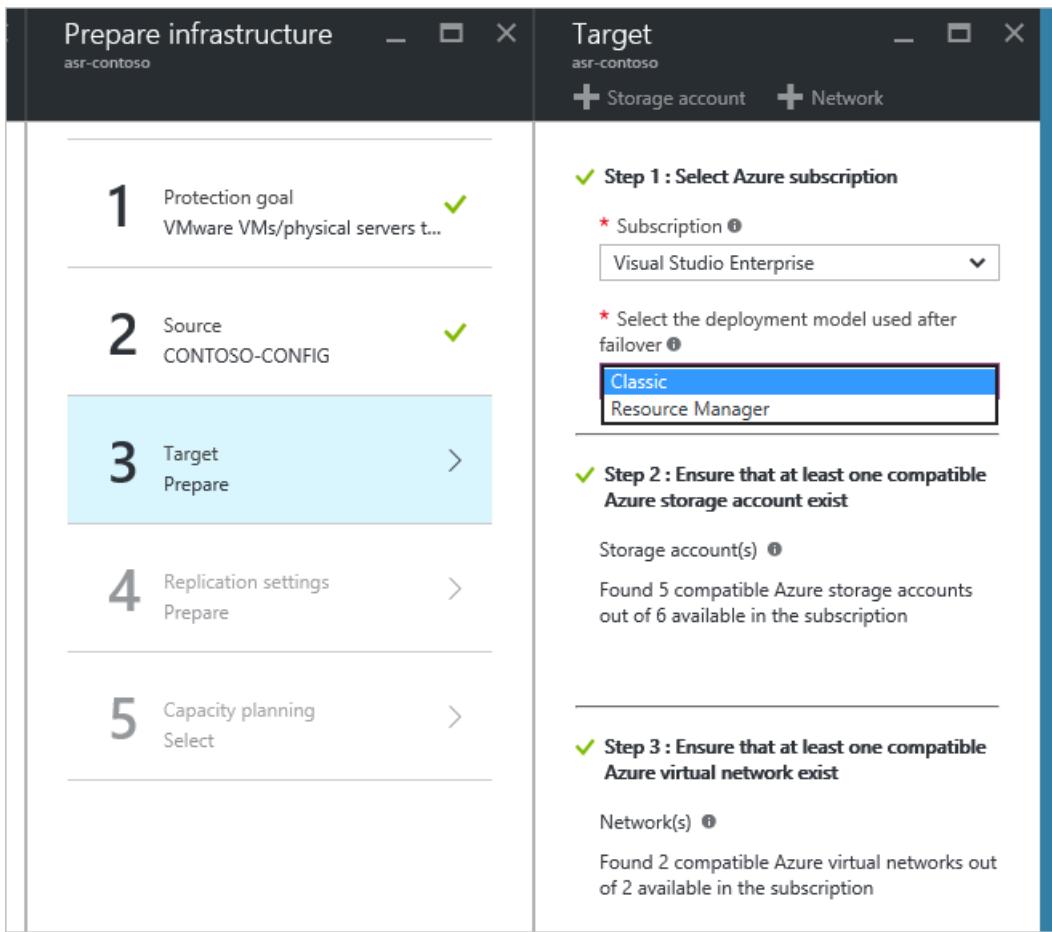
4. After the installation is complete, do the following in the **Microsoft Azure Site Recovery Configuration Server** window:

- Use the **Manage Accounts** tab to create the account that Site Recovery can use for automatic discovery. (In the scenario about protecting physical machines, setting up the account isn't relevant, but you need at least one account to enable one of the following steps. In this case, you can name the account and password as any.)
- Use the **Vault Registration** tab to upload the vault credential file.



#### Step 4: Set up the target environment

Select **Prepare infrastructure > Target**, and specify the deployment model that you want to use for VMs after failover. You can choose **Classic** or **Resource Manager**, depending on your scenario.



Site Recovery checks that you have one or more compatible Azure storage accounts and networks.

#### NOTE

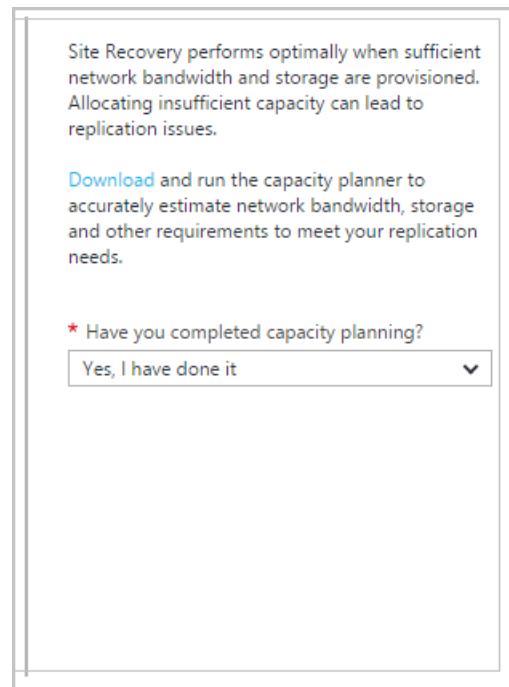
If you're using a premium storage account for replicated data, you need to set up an additional standard storage account to store replication logs.

#### Step 5: Set up replication settings

To verify that your configuration server is successfully associated with the replication policy that you create, follow [Set up replication settings](#).

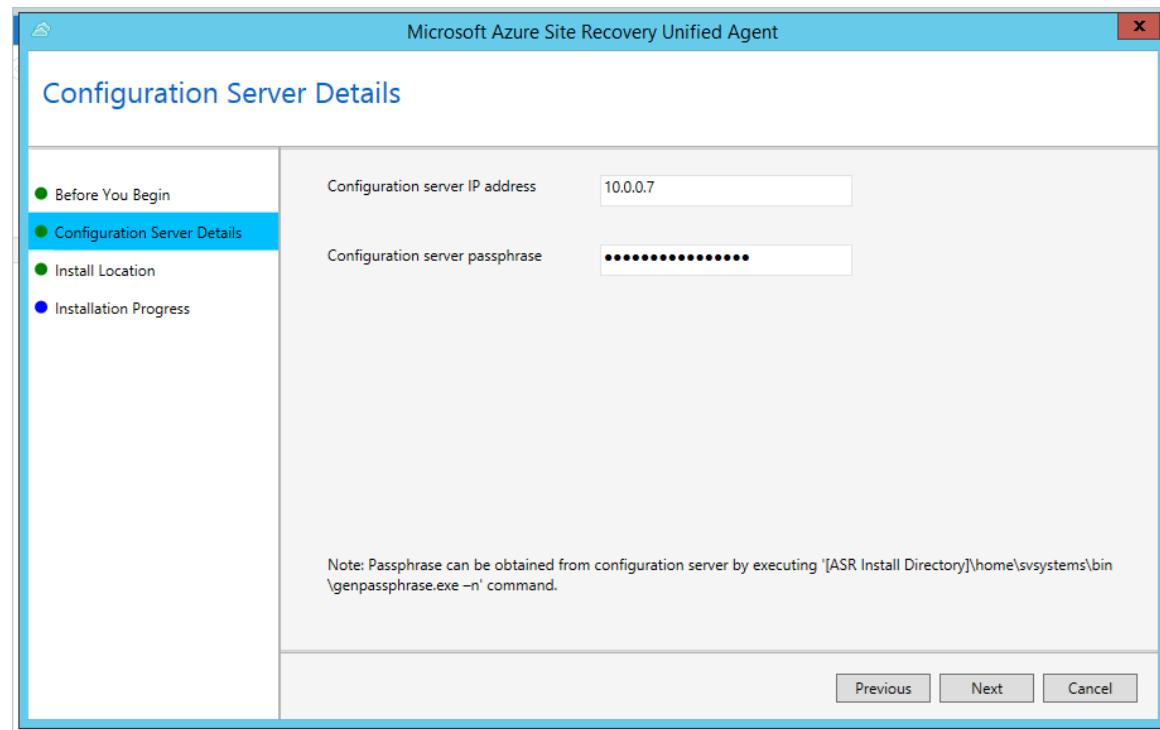
#### Step 6: Plan capacity

1. Use the [capacity planner](#) to accurately estimate network bandwidth, storage, and other requirements to meet your replication needs.
2. When you're done, select **Yes, I have done it** in **Have you completed capacity planning?**.



### Step 7: Install the mobility service and enable replication

1. You can choose to [push installation](#) to your source VMs or to [manually install the mobility service](#) on your source VMs. You can find the requirement of pushing installation and the path of the manual installer in the provided link. If you're doing a manual installation, you might need to use an internal IP address to find the configuration server.



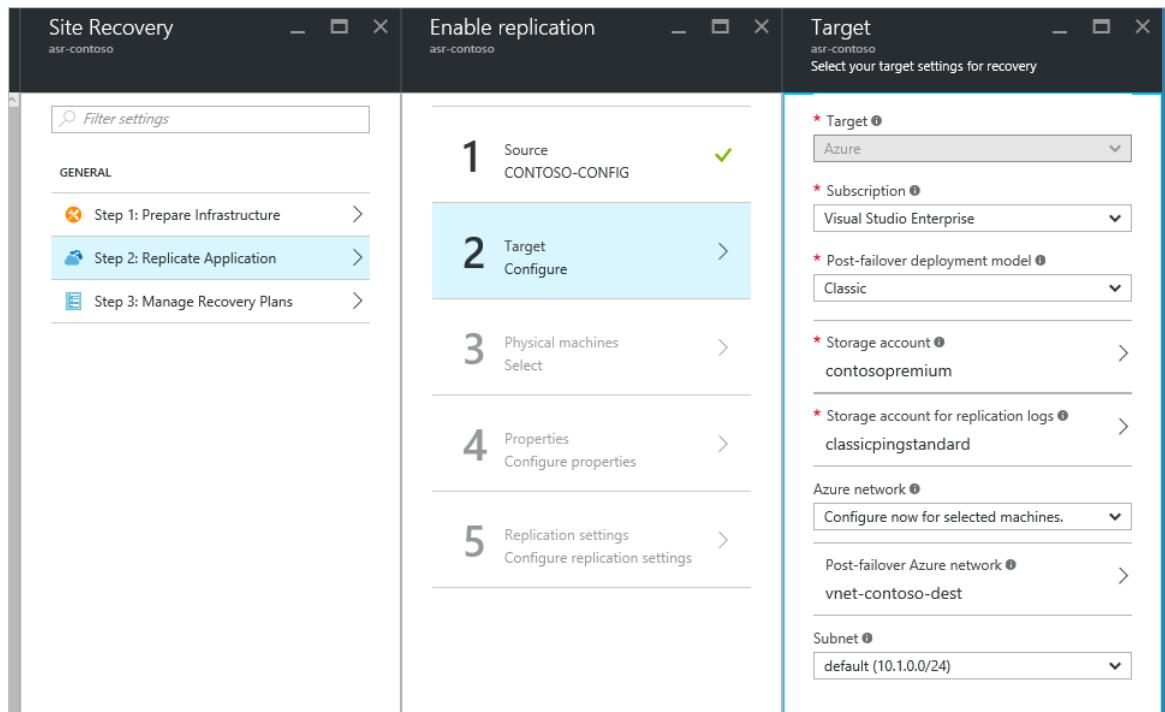
The failed-over VM will have two temporary disks: one from the primary VM and the other created during the provisioning of the VM in the recovery region. To exclude the temporary disk before replication, install the mobility service before you enable replication. To learn more about how to exclude the temporary disk, see [Exclude disks from replication](#).

2. Enable replication as follows:
  - a. Select **Replicate Application > Source**. After you've enabled replication for the first time, select **+ Replicate** in the vault to enable replication for additional machines.
  - b. In step 1, set up **Source** as your process server.

- c. In step 2, specify the post-failover deployment model, a premium storage account to migrate to, a standard storage account to save logs, and a virtual network to fail to.
- d. In step 3, add protected VMs by IP address. (You might need an internal IP address to find them.)
- e. In step 4, configure the properties by selecting the accounts that you set up previously on the process server.
- f. In step 5, choose the replication policy that you created previously in "Step 5: Set up replication settings."
- g. Select OK.

#### **NOTE**

When an Azure VM is deallocated and started again, there is no guarantee that it will get the same IP address. If the IP address of the configuration server/process server or the protected Azure VMs changes, the replication in this scenario might not work correctly.



When you design your Azure Storage environment, we recommend that you use separate storage accounts for each VM in an availability set. We recommend that you follow the best practice in the storage layer to [use multiple storage accounts for each availability set](#). Distributing VM disks to multiple storage accounts helps to improve storage availability and distributes the I/O across the Azure storage infrastructure.

If your VMs are in an availability set, instead of replicating disks of all VMs into one storage account, we highly recommend migrating multiple VMs multiple times. That way, the VMs in the same availability set do not share a single storage account. Use the **Enable Replication** pane to set up a destination storage account for each VM, one at a time.

You can choose a post-failover deployment model according to your need. If you choose Azure Resource Manager as your post-failover deployment model, you can fail over a VM (Resource Manager) to a VM (Resource Manager), or you can fail over a VM (classic) to a VM (Resource Manager).

#### **Step 8: Run a test failover**

To check whether your replication is complete, select your Site Recovery instance and then select **Settings > Replicated Items**. You will see the status and percentage of your replication process.

After initial replication is complete, run a test failover to validate your replication strategy. For detailed steps of a test failover, see [Run a test failover in Site Recovery](#).

#### **NOTE**

Before you run any failover, make sure that your VMs and replication strategy meet the requirements. For more information about running a test failover, see [Test failover to Azure in Site Recovery](#).

You can see the status of your test failover in **Settings > Jobs > YOUR\_FAILOVER\_PLAN\_NAME**. In the pane, you can see a breakdown of the steps and success/failure results. If the test failover fails at any step, select the step to check the error message.

#### **Step 9: Run a failover**

After the test failover is completed, run a failover to migrate your disks to Premium Storage and replicate the VM instances. Follow the detailed steps in [Run a failover](#).

Be sure to select **Shut down VMs and synchronize the latest data**. This option specifies that Site Recovery should try to shut down the protected VMs and synchronize the data so that the latest version of the data will be failed over. If you don't select this option or the attempt doesn't succeed, the failover will be from the latest available recovery point for the VM.

Site Recovery will create a VM instance whose type is the same as or similar to a Premium Storage-capable VM. You can check the performance and price of various VM instances by going to [Windows Virtual Machines Pricing](#) or [Linux Virtual Machines Pricing](#).

## Post-migration steps

1. **Configure replicated VMs to the availability set if applicable.** Site Recovery does not support migrating VMs along with the availability set. Depending on the deployment of your replicated VM, do one of the following:
  - For a VM created through the classic deployment model: Add the VM to the availability set in the Azure portal. For detailed steps, go to [Add an existing virtual machine to an availability set](#).
  - For a VM created through the Resource Manager deployment model: Save your configuration of the VM and then delete and re-create the VMs in the availability set. To do so, use the script at [Set Azure Resource Manager VM Availability Set](#). Before you run this script, check its limitations and plan your downtime.
2. **Delete old VMs and disks.** Make sure that the Premium disks are consistent with source disks and that the new VMs perform the same function as the source VMs. Delete the VM and delete the disks from your source storage accounts in the Azure portal. If there's a problem in which the disk is not deleted even though you deleted the VM, see [Troubleshoot storage resource deletion errors](#).
3. **Clean the Azure Site Recovery infrastructure.** If Site Recovery is no longer needed, you can clean its infrastructure. Delete replicated items, the configuration server, and the recovery policy, and then delete the Azure Site Recovery vault.

## Troubleshooting

- [Monitor and troubleshoot protection for virtual machines and physical servers](#)
- [Microsoft Q&A question page for Microsoft Azure Site Recovery](#)

## Next steps

For specific scenarios for migrating virtual machines, see the following resources:

- [Migrate Azure Virtual Machines between Storage Accounts](#)
- [Upload a Linux virtual hard disk](#)

- Migrating Virtual Machines from Amazon AWS to Microsoft Azure

Also, see the following resources to learn more about Azure Storage and Azure Virtual Machines:

- [Azure Storage](#)
- [Azure Virtual Machines](#)
- [Select a disk type for IaaS VMs](#)

# Migrate to Premium Storage by using Azure Site Recovery

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

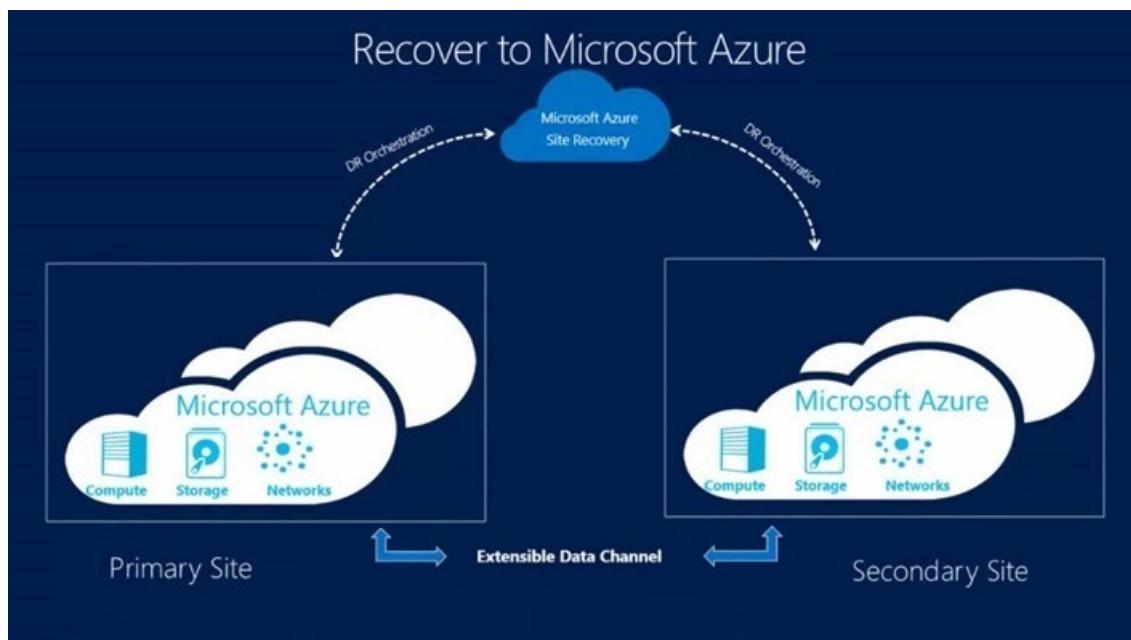
Azure premium SSDs deliver high-performance, low-latency disk support for virtual machines (VMs) that are running I/O-intensive workloads. This guide helps you migrate your VM disks from a standard storage account to a premium storage account by using [Azure Site Recovery](#).

Site Recovery is an Azure service that contributes to your strategy for business continuity and disaster recovery by orchestrating the replication of on-premises physical servers and VMs to the cloud (Azure) or to a secondary datacenter. When outages occur in your primary location, you fail over to the secondary location to keep applications and workloads available. You fail back to your primary location when it returns to normal operation.

Site Recovery provides test failovers to support disaster recovery drills without affecting production environments. You can run failovers with minimal data loss (depending on replication frequency) for unexpected disasters. In the scenario of migrating to Premium Storage, you can use the [failover in Site Recovery](#) to migrate target disks to a premium storage account.

We recommend migrating to Premium Storage by using Site Recovery because this option provides minimal downtime. This option also avoids the manual execution of copying disks and creating new VMs. Site Recovery will systematically copy your disks and create new VMs during failover.

Site Recovery supports a number of types of failover with minimal or no downtime. To plan your downtime and estimate data loss, see the [types of failover in Site Recovery](#). If you [prepare to connect to Azure VMs after failover](#), you should be able to connect to the Azure VM by using RDP after failover.



## Azure Site Recovery components

These Site Recovery components are relevant to this migration scenario:

- **Configuration server** is an Azure VM that coordinates communication and manages data replication and recovery processes. On this VM, you run a single setup file to install the configuration server and an

additional component, called a process server, as a replication gateway. Read about [configuration server prerequisites](#). You set up the configuration server only once, and you can use it for all migrations to the same region.

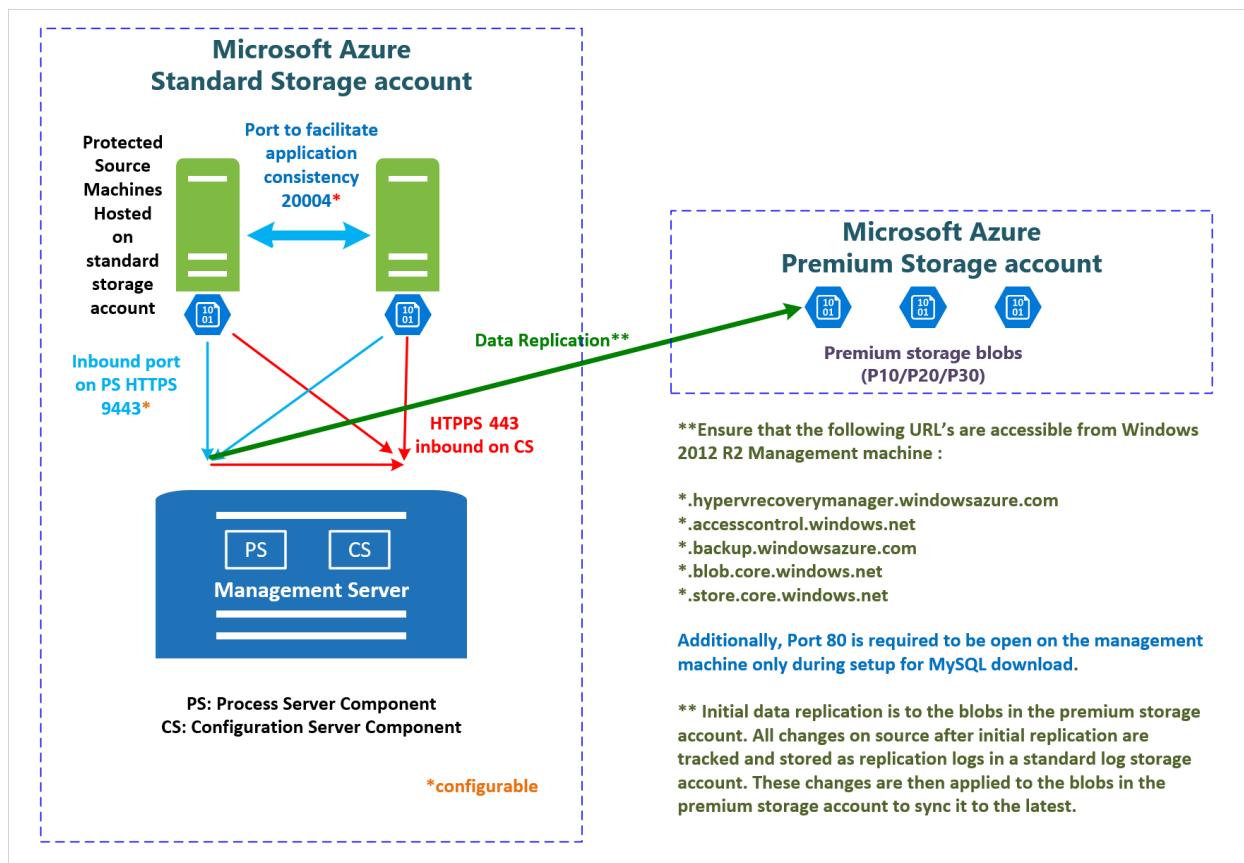
- **Process server** is a replication gateway that:

1. Receives replication data from source VMs.
2. Optimizes the data with caching, compression, and encryption.
3. Sends the data to a storage account.

It also handles push installation of the mobility service to source VMs and performs automatic discovery of source VMs. The default process server is installed on the configuration server. You can deploy additional standalone process servers to scale your deployment. Read about [best practices for process server deployment](#) and [deploying additional process servers](#). You set up the process server only once, and you can use it for all migrations to the same region.

- **Mobility service** is a component that is deployed on every standard VM that you want to replicate. It captures data writes on the standard VM and forwards them to the process server. Read about [replicated machine prerequisites](#).

This graphic shows how these components interact:



#### NOTE

Site Recovery does not support the migration of Storage Spaces disks.

For additional components for other scenarios, see [Scenario architecture](#).

## Azure essentials

These are the Azure requirements for this migration scenario:

- An Azure subscription.
- An Azure premium storage account to store replicated data.
- An Azure virtual network to which VMs will connect when they're created at failover. The Azure virtual network must be in the same region as the one in which Site Recovery runs.
- An Azure standard storage account to store replication logs. This can be the same storage account for the VM disks that are being migrated.

## Prerequisites

- Understand the relevant migration scenario components in the preceding section.
- Plan your downtime by learning about [failover in Site Recovery](#).

## Setup and migration steps

You can use Site Recovery to migrate Azure IaaS VMs between regions or within same region. The following instructions are tailored for this migration scenario from the article [Replicate VMware VMs or physical servers to Azure](#). Please follow the links for detailed steps in addition to the instructions in this article.

### Step 1: Create a Recovery Services vault

1. Open the [Azure portal](#).
2. Select **Create a resource > Management > Backup and Site Recovery (OMS)**. Alternatively, you can select **Browse > Recovery Services Vault > Add**.

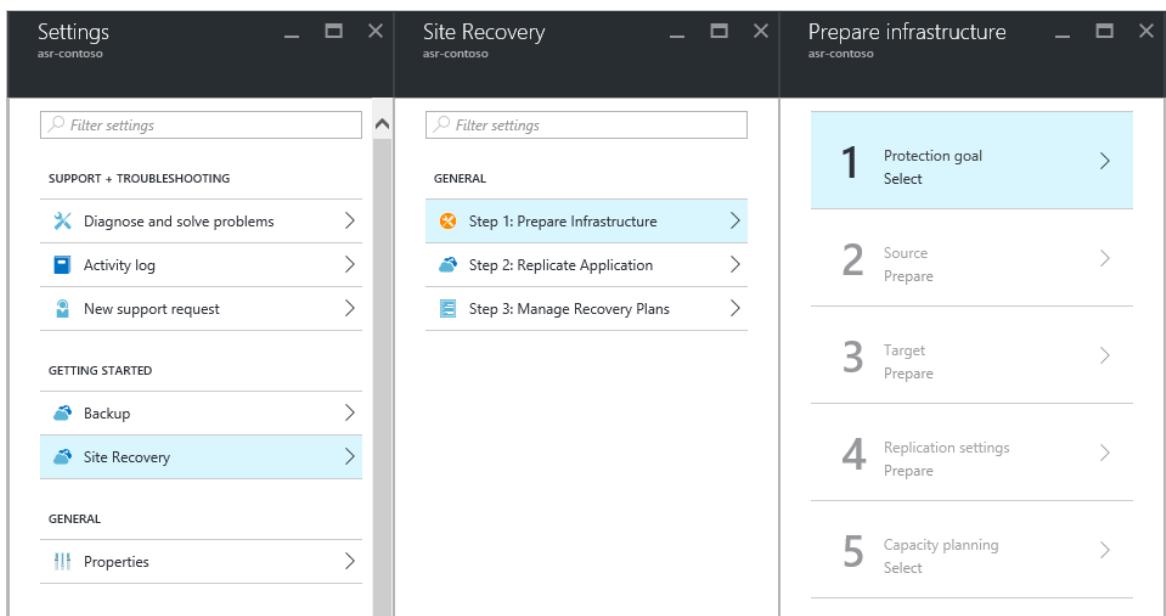
#### NOTE

Backup and Site Recovery was formerly part of the [OMS suite](#).

3. Specify a region that VMs will be replicated to. For the purpose of migration in the same region, select the region where your source VMs and source storage accounts are.

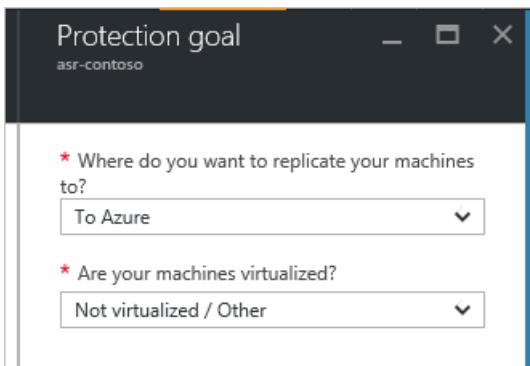
### Step 2: Choose your protection goals

1. On the VM where you want to install the configuration server, open the [Azure portal](#).
2. Go to **Recovery Services vaults > Settings > Site Recovery > Step 1: Prepare Infrastructure > Protection goal**.



3. Under **Protection goal**, in the first drop-down list, select **To Azure**. In the second drop-down list, select

Not virtualized / Other, and then select OK.



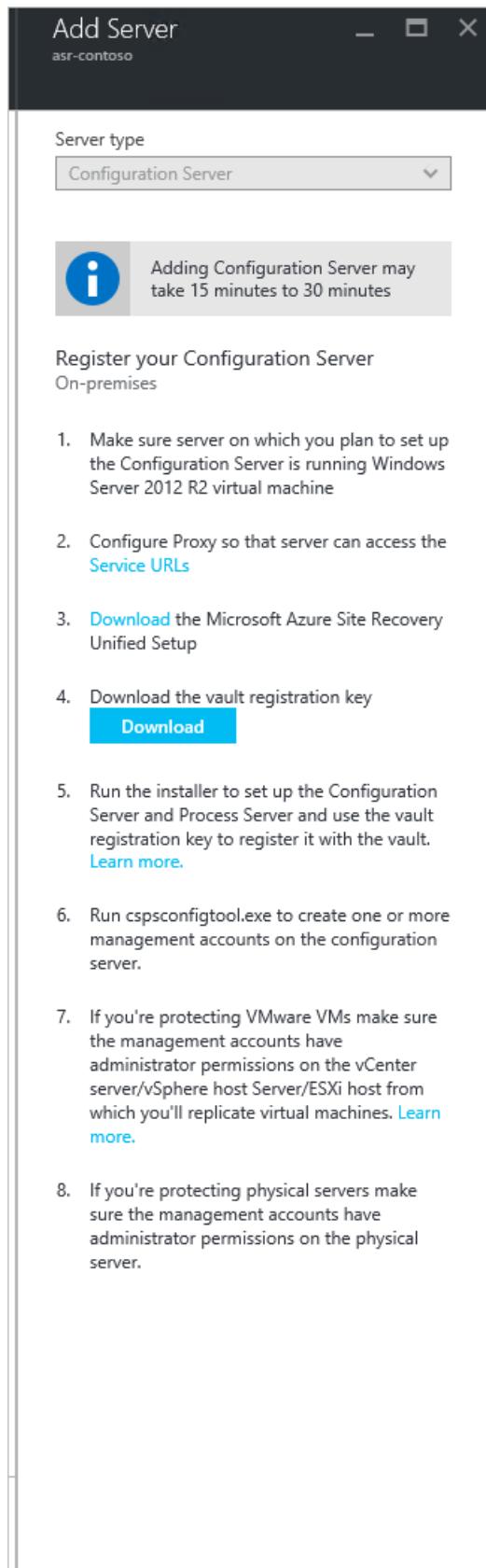
### Step 3: Set up the source environment (configuration server)

1. Download Azure Site Recovery Unified Setup and the vault registration key by going to the **Prepare infrastructure** > **Prepare source** > **Add Server** panes.

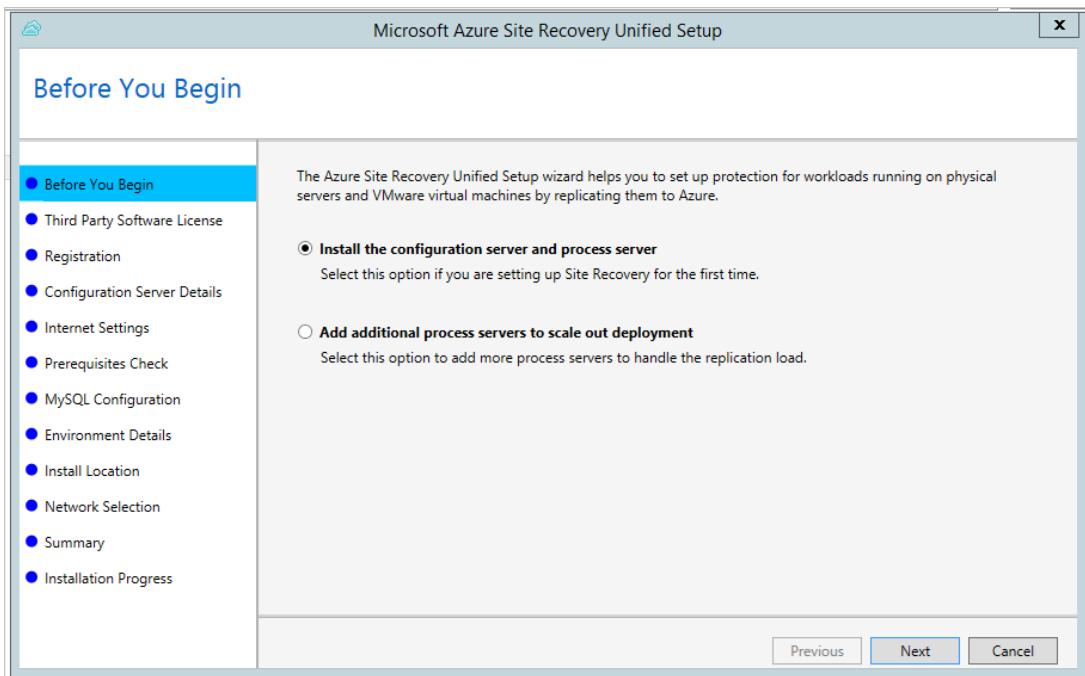
You will need the vault registration key to run the unified setup. The key is valid for five days after you generate it.

The screenshot displays two side-by-side windows. The left window is 'Prepare infrastructure' for 'asr-contoso', showing a numbered list from 1 to 5: 1. Protection goal (checkmark), 2. Source Prepare, 3. Target Prepare, 4. Replication settings, and 5. Capacity planning. The 'Source Prepare' step is highlighted with a blue background. The right window is 'Prepare source' for 'asr-contoso', titled '+ Configuration...'. It contains a section titled '→ Step 1 : Select Configuration Server' with a note: '(0 servers found) Click on +Configuration Server in the command bar above to setup one on your source environment and register it with this vault.' There is also an orange wrench icon.

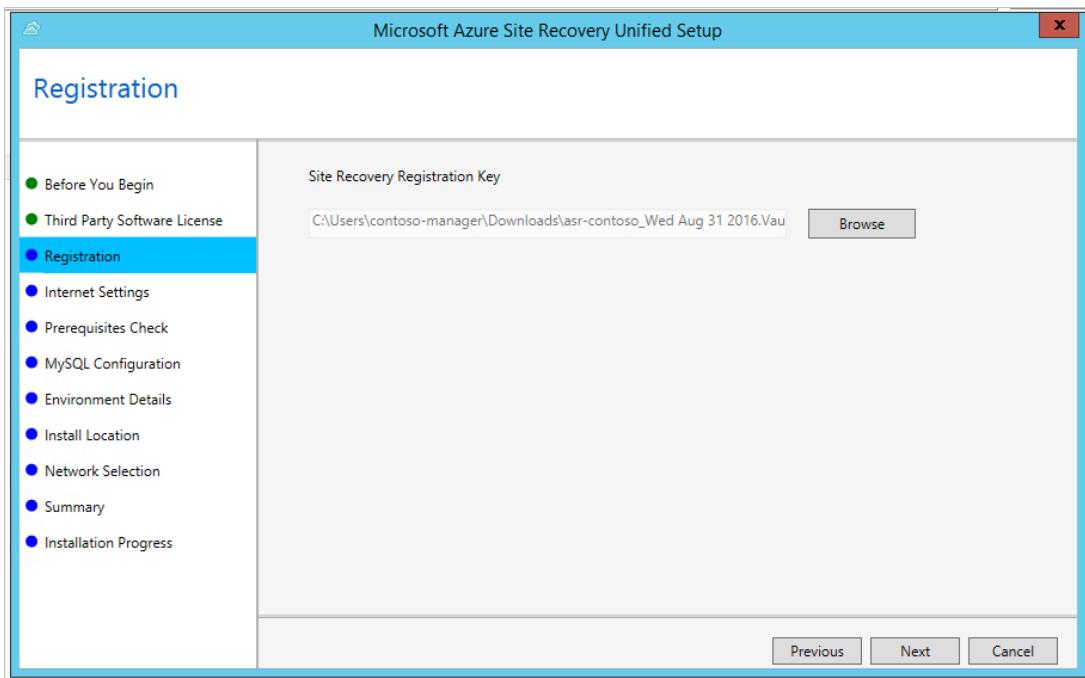
2. In the **Add Server** pane, add a configuration server.



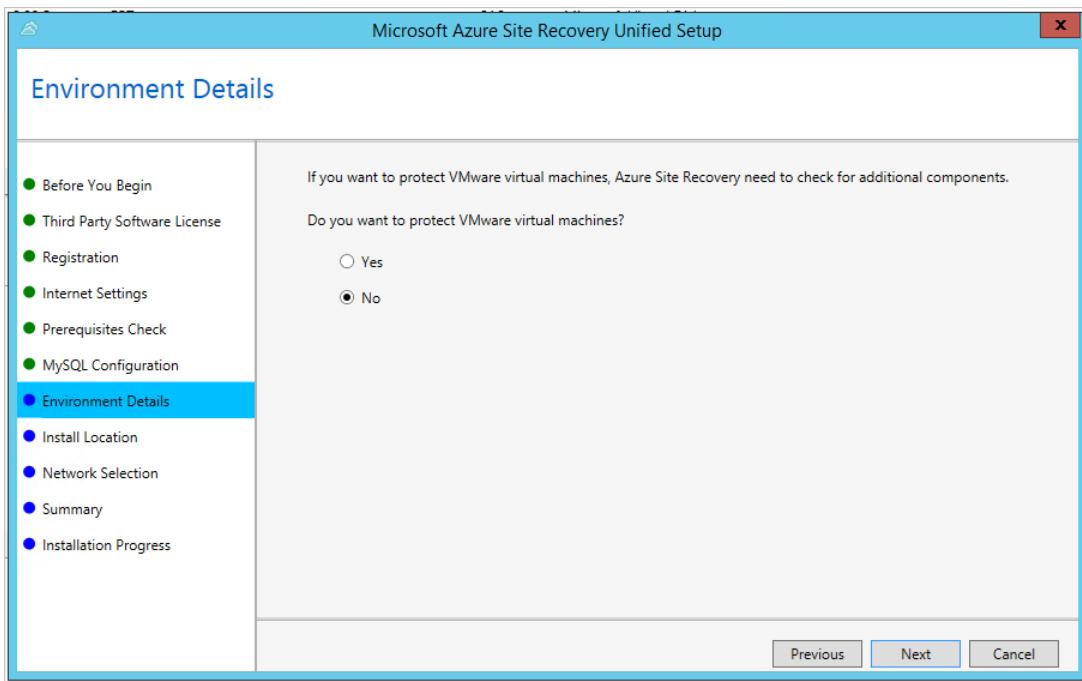
3. On the VM that you're using as the configuration server, run Unified Setup to install the configuration server and the process server. You can [walk through the screenshots](#) to complete the installation. You can refer to the following screenshots for steps specified for this migration scenario.
  - a. In **Before You Begin**, select **Install the configuration server and process server**.



- b. In **Registration**, browse and select the registration key that you downloaded from the vault.

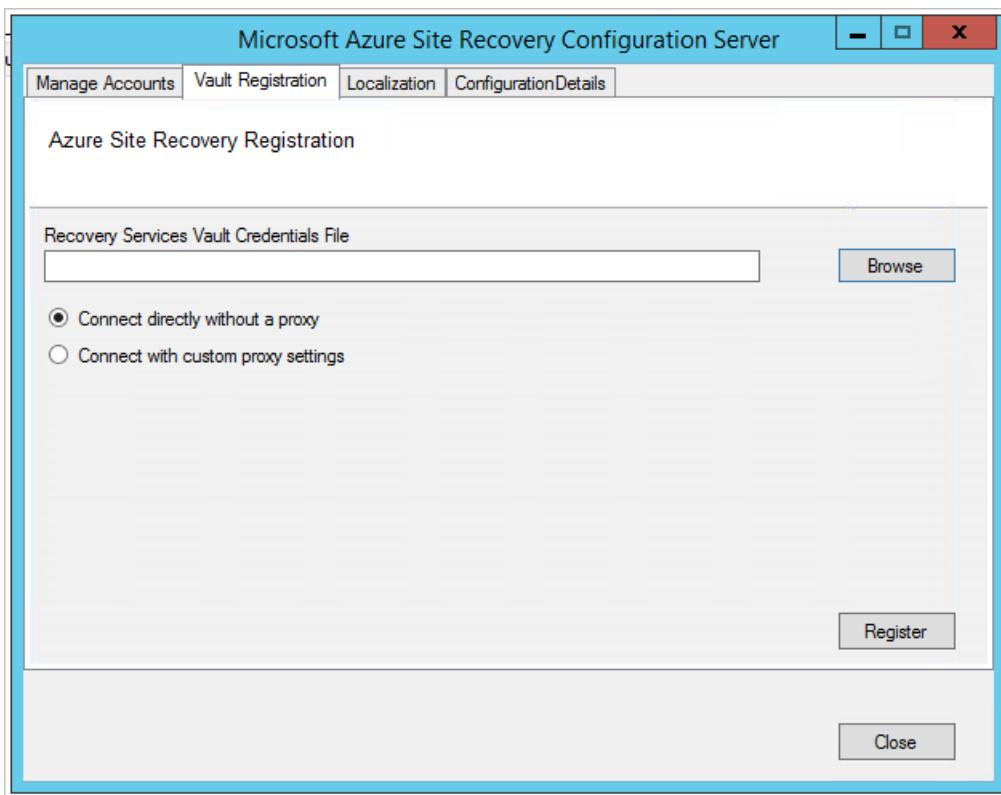


- c. In **Environment Details**, select whether you're going to replicate VMware VMs. For this migration scenario, choose **No**.



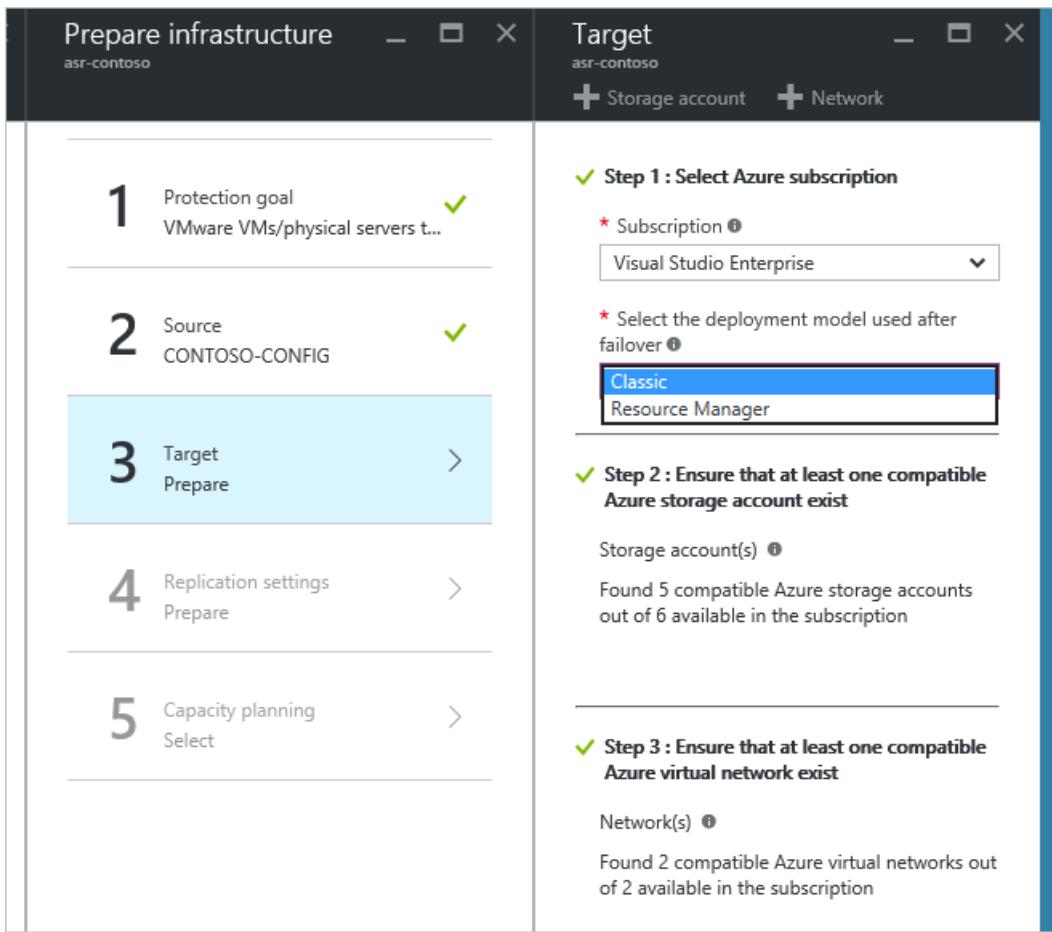
4. After the installation is complete, do the following in the **Microsoft Azure Site Recovery Configuration Server** window:

- Use the **Manage Accounts** tab to create the account that Site Recovery can use for automatic discovery. (In the scenario about protecting physical machines, setting up the account isn't relevant, but you need at least one account to enable one of the following steps. In this case, you can name the account and password as any.)
- Use the **Vault Registration** tab to upload the vault credential file.



#### Step 4: Set up the target environment

Select **Prepare infrastructure > Target**, and specify the deployment model that you want to use for VMs after failover. You can choose **Classic** or **Resource Manager**, depending on your scenario.



Site Recovery checks that you have one or more compatible Azure storage accounts and networks.

#### NOTE

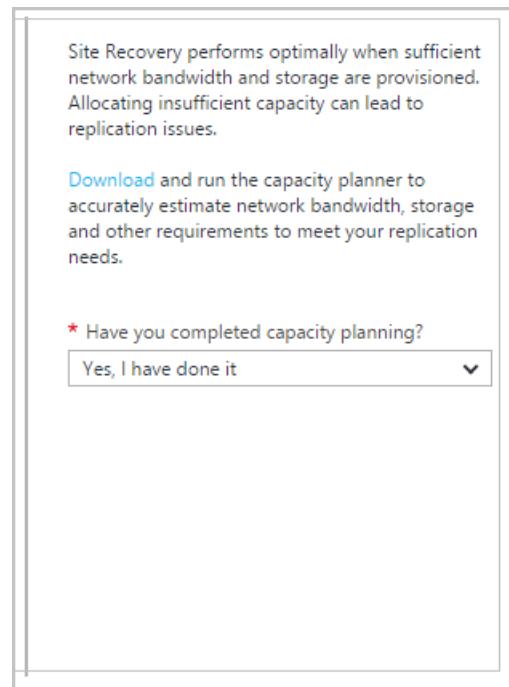
If you're using a premium storage account for replicated data, you need to set up an additional standard storage account to store replication logs.

#### Step 5: Set up replication settings

To verify that your configuration server is successfully associated with the replication policy that you create, follow [Set up replication settings](#).

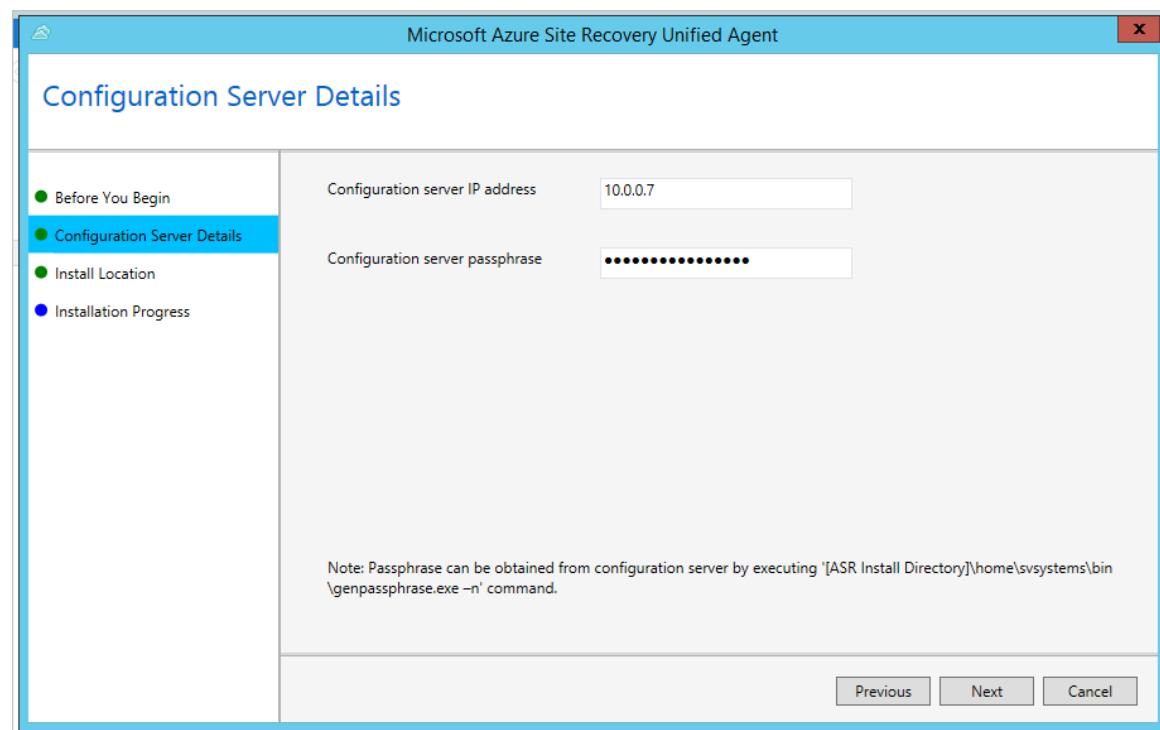
#### Step 6: Plan capacity

1. Use the [capacity planner](#) to accurately estimate network bandwidth, storage, and other requirements to meet your replication needs.
2. When you're done, select **Yes, I have done it** in **Have you completed capacity planning?**.



### Step 7: Install the mobility service and enable replication

1. You can choose to [push installation](#) to your source VMs or to [manually install the mobility service](#) on your source VMs. You can find the requirement of pushing installation and the path of the manual installer in the provided link. If you're doing a manual installation, you might need to use an internal IP address to find the configuration server.



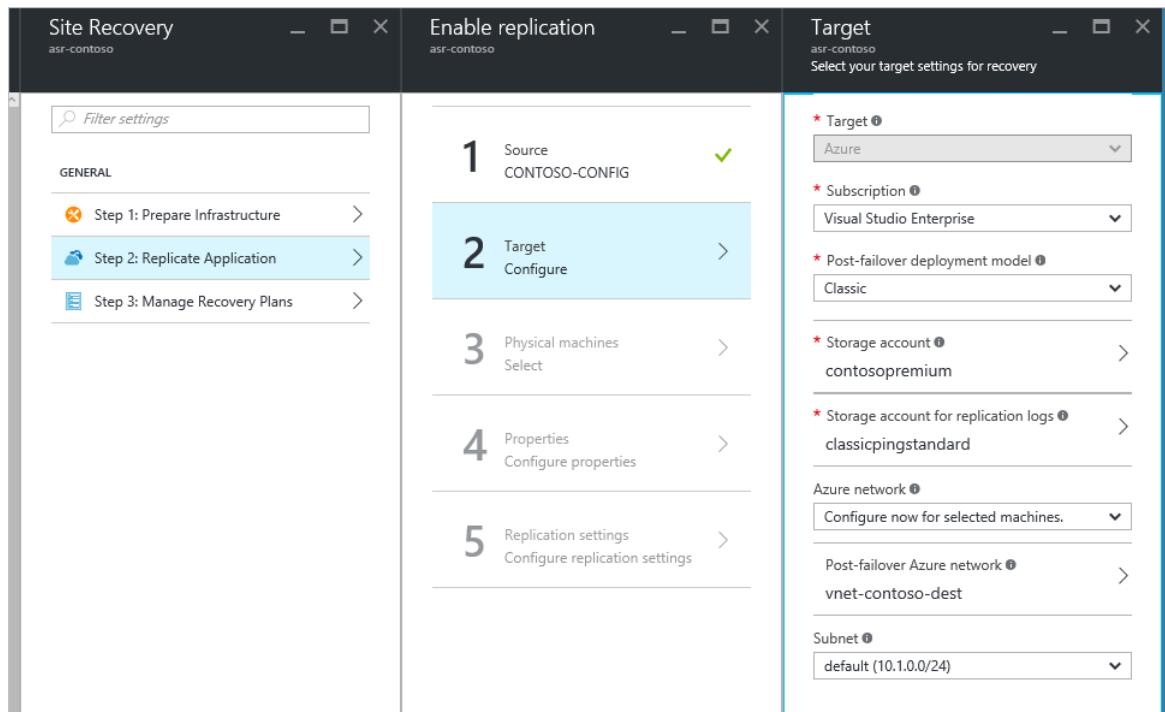
The failed-over VM will have two temporary disks: one from the primary VM and the other created during the provisioning of the VM in the recovery region. To exclude the temporary disk before replication, install the mobility service before you enable replication. To learn more about how to exclude the temporary disk, see [Exclude disks from replication](#).

2. Enable replication as follows:
  - a. Select **Replicate Application > Source**. After you've enabled replication for the first time, select **+ Replicate** in the vault to enable replication for additional machines.
  - b. In step 1, set up **Source** as your process server.

- c. In step 2, specify the post-failover deployment model, a premium storage account to migrate to, a standard storage account to save logs, and a virtual network to fail to.
- d. In step 3, add protected VMs by IP address. (You might need an internal IP address to find them.)
- e. In step 4, configure the properties by selecting the accounts that you set up previously on the process server.
- f. In step 5, choose the replication policy that you created previously in "Step 5: Set up replication settings."
- g. Select OK.

#### **NOTE**

When an Azure VM is deallocated and started again, there is no guarantee that it will get the same IP address. If the IP address of the configuration server/process server or the protected Azure VMs changes, the replication in this scenario might not work correctly.



When you design your Azure Storage environment, we recommend that you use separate storage accounts for each VM in an availability set. We recommend that you follow the best practice in the storage layer to [use multiple storage accounts for each availability set](#). Distributing VM disks to multiple storage accounts helps to improve storage availability and distributes the I/O across the Azure storage infrastructure.

If your VMs are in an availability set, instead of replicating disks of all VMs into one storage account, we highly recommend migrating multiple VMs multiple times. That way, the VMs in the same availability set do not share a single storage account. Use the **Enable Replication** pane to set up a destination storage account for each VM, one at a time.

You can choose a post-failover deployment model according to your need. If you choose Azure Resource Manager as your post-failover deployment model, you can fail over a VM (Resource Manager) to a VM (Resource Manager), or you can fail over a VM (classic) to a VM (Resource Manager).

#### **Step 8: Run a test failover**

To check whether your replication is complete, select your Site Recovery instance and then select **Settings > Replicated Items**. You will see the status and percentage of your replication process.

After initial replication is complete, run a test failover to validate your replication strategy. For detailed steps of a test failover, see [Run a test failover in Site Recovery](#).

#### **NOTE**

Before you run any failover, make sure that your VMs and replication strategy meet the requirements. For more information about running a test failover, see [Test failover to Azure in Site Recovery](#).

You can see the status of your test failover in **Settings > Jobs > YOUR\_FAILOVER\_PLAN\_NAME**. In the pane, you can see a breakdown of the steps and success/failure results. If the test failover fails at any step, select the step to check the error message.

#### **Step 9: Run a failover**

After the test failover is completed, run a failover to migrate your disks to Premium Storage and replicate the VM instances. Follow the detailed steps in [Run a failover](#).

Be sure to select **Shut down VMs and synchronize the latest data**. This option specifies that Site Recovery should try to shut down the protected VMs and synchronize the data so that the latest version of the data will be failed over. If you don't select this option or the attempt doesn't succeed, the failover will be from the latest available recovery point for the VM.

Site Recovery will create a VM instance whose type is the same as or similar to a Premium Storage-capable VM. You can check the performance and price of various VM instances by going to [Windows Virtual Machines Pricing](#) or [Linux Virtual Machines Pricing](#).

## Post-migration steps

1. **Configure replicated VMs to the availability set if applicable.** Site Recovery does not support migrating VMs along with the availability set. Depending on the deployment of your replicated VM, do one of the following:
  - For a VM created through the classic deployment model: Add the VM to the availability set in the Azure portal. For detailed steps, go to [Add an existing virtual machine to an availability set](#).
  - For a VM created through the Resource Manager deployment model: Save your configuration of the VM and then delete and re-create the VMs in the availability set. To do so, use the script at [Set Azure Resource Manager VM Availability Set](#). Before you run this script, check its limitations and plan your downtime.
2. **Delete old VMs and disks.** Make sure that the Premium disks are consistent with source disks and that the new VMs perform the same function as the source VMs. Delete the VM and delete the disks from your source storage accounts in the Azure portal. If there's a problem in which the disk is not deleted even though you deleted the VM, see [Troubleshoot storage resource deletion errors](#).
3. **Clean the Azure Site Recovery infrastructure.** If Site Recovery is no longer needed, you can clean its infrastructure. Delete replicated items, the configuration server, and the recovery policy, and then delete the Azure Site Recovery vault.

## Troubleshooting

- [Monitor and troubleshoot protection for virtual machines and physical servers](#)
- [Microsoft Q&A question page for Microsoft Azure Site Recovery](#)

## Next steps

For specific scenarios for migrating virtual machines, see the following resources:

- [Migrate Azure Virtual Machines between Storage Accounts](#)
- [Create and upload a Windows Server VHD to Azure](#)

- Migrating Virtual Machines from Amazon AWS to Microsoft Azure

Also, see the following resources to learn more about Azure Storage and Azure Virtual Machines:

- [Azure Storage](#)
- [Azure Virtual Machines](#)

# Migrate Azure VMs to Managed Disks in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

Azure Managed Disks simplifies your storage management by removing the need to separately manage storage accounts. You can also migrate your existing Azure VMs to Managed Disks to benefit from better reliability of VMs in an Availability Set. It ensures that the disks of different VMs in an Availability Set are sufficiently isolated from each other to avoid single point of failures. It automatically places disks of different VMs in an Availability Set in different Storage scale units (stamps) which limits the impact of single Storage scale unit failures caused due to hardware and software failures. Based on your needs, you can choose from four types of storage options. To learn about the available disk types, see our article [Select a disk type](#)

## Migration scenarios

You can migrate to Managed Disks in following scenarios:

SCENARIO	ARTICLE
Convert stand alone VMs and VMs in an availability set to managed disks	<a href="#">Convert VMs to use managed disks</a>
Convert a single VM from classic to Resource Manager on managed disks	<a href="#">Create a VM from a classic VHD</a>
Convert all the VMs in a vNet from classic to Resource Manager on managed disks	<a href="#">Migrate IaaS resources from classic to Resource Manager</a> and then <a href="#">Convert a VM from unmanaged disks to managed disks</a>
Upgrade VMs with standard unmanaged disks to VMs with managed premium disks	First, <a href="#">Convert a Windows virtual machine from unmanaged disks to managed disks</a> . Then <a href="#">Update the storage type of a managed disk</a> .

### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

## Next steps

- Learn more about [Managed Disks](#)
- Review the [pricing for Managed Disks](#).

# Migrate a Linux virtual machine from unmanaged disks to managed disks

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

If you have existing Linux virtual machines (VMs) that use unmanaged disks, you can migrate the VMs to use [Azure Managed Disks](#). This process converts both the OS disk and any attached data disks.

This article shows you how to migrate VMs by using the Azure CLI. If you need to install or upgrade it, see [Install Azure CLI](#).

## Before you begin

- Review [the FAQ about migration to Managed Disks](#).
- The migration will restart the VM, so schedule the migration of your VMs during a pre-existing maintenance window.
- The migration isn't reversible.
- Any users with the [Virtual Machine Contributor](#) role won't be able to change the VM size (as they could pre-migration). This is because VMs with managed disks require the user to have the Microsoft.Compute/disks/write permission on the OS disks.
- Be sure to test the migration. Migrate a test virtual machine before you perform the migration in production.
- During the migration, you deallocate the VM. The VM receives a new IP address when it's started after the migration. If needed, you can [assign a static IP address](#) to the VM.
- Review the minimum version of the Azure VM agent required to support the migration process. For information on how to check and update your agent version, see [Minimum version support for VM agents in Azure](#)
- The original VHDs and the storage account used by the VM before migration are not deleted. They continue to incur charges. To avoid being billed for these artifacts, delete the original VHD blobs after you verify that the migration is complete. If you need to find these unattached disks in order to delete them, see our article [Find and delete unattached Azure managed and unmanaged disks](#).

## Migrate single-instance VMs

This section covers how to migrate single-instance Azure VMs from unmanaged disks to managed disks. (If your VMs are in an availability set, see the next section.) You can use this process to migrate the VMs from premium (SSD) unmanaged disks to premium managed disks, or from standard (HDD) unmanaged disks to standard managed disks.

1. Deallocate the VM by using `az vm deallocate`. The following example deallocates the VM named `myVM` in the resource group named `myResourceGroup`:

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

2. Migrate the VM to managed disks by using [az vm convert](#). The following process converts the VM named `myVM`, including the OS disk and any data disks:

```
az vm convert --resource-group myResourceGroup --name myVM
```

3. Start the VM after the migration to managed disks by using [az vm start](#). The following example starts the VM named `myVM` in the resource group named `myResourceGroup`.

```
az vm start --resource-group myResourceGroup --name myVM
```

## Migrate VMs in an availability set

If the VMs that you want to migrate to managed disks are in an availability set, you first need to migrate the availability set to a managed availability set.

All VMs in the availability set must be deallocated before you migrate the availability set. Plan to migrate all VMs to managed disks after the availability set itself has been converted to a managed availability set. Then, start all the VMs and continue operating as normal.

1. List all VMs in an availability set by using [az vm availability-set list](#). The following example lists all VMs in the availability set named `myAvailabilitySet` in the resource group named `myResourceGroup`:

```
az vm availability-set show \
--resource-group myResourceGroup \
--name myAvailabilitySet \
--query [virtualMachines[*].id] \
--output table
```

2. Deallocate all the VMs by using [az vm deallocate](#). The following example deallocates the VM named `myVM` in the resource group named `myResourceGroup`:

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

3. Migrate the availability set by using [az vm availability-set convert](#). The following example converts the availability set named `myAvailabilitySet` in the resource group named `myResourceGroup`:

```
az vm availability-set convert \
--resource-group myResourceGroup \
--name myAvailabilitySet
```

4. Migrate all the VMs to managed disks by using [az vm convert](#). The following process converts the VM named `myVM`, including the OS disk and any data disks:

```
az vm convert --resource-group myResourceGroup --name myVM
```

5. Start all the VMs after the migration to managed disks by using [az vm start](#). The following example starts the VM named `myVM` in the resource group named `myResourceGroup`:

```
az vm start --resource-group myResourceGroup --name myVM
```

## Migrate using the Azure portal

You can also migrate unmanaged disks to managed disks using the Azure portal.

1. Sign in to the [Azure portal](#).
2. Select the VM from the list of VMs in the portal.
3. In the blade for the VM, select **Disks** from the menu.
4. At the top of the **Disks** blade, select **Migrate to managed disks**.
5. If your VM is in an availability set, there will be a warning on the **Migrate to managed disks** blade that you need to migrate the availability set first. The warning should have a link you can click to migrate the availability set. Once the availability set is converted or if your VM is not in an availability set, click **Migrate** to start the process of migrating your disks to managed disks.

The VM will be stopped and restarted after migration is complete.

## Next steps

For more information about storage options, see [Azure Managed Disks overview](#).

# Migrate a Windows virtual machine from unmanaged disks to managed disks

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

If you have existing Windows virtual machines (VMs) that use unmanaged disks, you can migrate the VMs to use managed disks through the [Azure Managed Disks](#) service. This process converts both the operating system (OS) disk and any attached data disks.

## Before you begin

- Review [Plan for the migration to Managed Disks](#).
- Review [the FAQ about migration to Managed Disks](#).
- The migration will restart the VM, so schedule the migration of your VMs during a pre-existing maintenance window.
- The migration isn't reversible.
- Any users with the [Virtual Machine Contributor](#) role won't be able to change the VM size (as they could pre-migration). This is because VMs with managed disks require the user to have the Microsoft.Compute/disks/write permission on the OS disks.
- Be sure to test the migration. Migrate a test virtual machine before you perform the migration in production.
- During the migration, you deallocate the VM. The VM receives a new IP address when it's started after the migration. If needed, you can [assign a static IP address](#) to the VM.
- Review the minimum version of the Azure VM agent required to support the migration process. For information on how to check and update your agent version, see [Minimum version support for VM agents in Azure](#)
- The original VHDs and the storage account used by the VM before migration are not deleted. They continue to incur charges. To avoid being billed for these artifacts, delete the original VHD blobs after you verify that the migration is complete. If you need to find these unattached disks in order to delete them, see our article [Find and delete unattached Azure managed and unmanaged disks](#).

## Migrate single-instance VMs

This section covers how to migrate single-instance Azure VMs from unmanaged disks to managed disks. (If your VMs are in an availability set, see the next section.)

1. Deallocate the VM by using the [Stop-AzVM](#) cmdlet. The following example deallocates the VM named `myVM` in the resource group named `myResourceGroup`:

```
$rgName = "myResourceGroup"  
$vmName = "myVM"  
Stop-AzVM -ResourceGroupName $rgName -Name $vmName -Force
```

- Migrate the VM to managed disks by using the [ConvertTo-AzVMManagedDisk](#) cmdlet. The following process converts the previous VM, including the OS disk and any data disks, and starts the Virtual Machine:

```
ConvertTo-AzVMManagedDisk -ResourceGroupName $rgName -VMName $vmName
```

## Migrate VMs in an availability set

If the VMs that you want to migrate to managed disks are in an availability set, you first need to migrate the availability set to a managed availability set.

- Migrate the availability set by using the [Update-AzAvailabilitySet](#) cmdlet. The following example updates the availability set named `myAvailabilitySet` in the resource group named `myResourceGroup`:

```
$rgName = 'myResourceGroup'
$avSetName = 'myAvailabilitySet'

$avSet = Get-AzAvailabilitySet -ResourceGroupName $rgName -Name $avSetName
Update-AzAvailabilitySet -AvailabilitySet $avSet -Sku Aligned
```

If the region where your availability set is located has only 2 managed fault domains but the number of unmanaged fault domains is 3, this command shows an error similar to "The specified fault domain count 3 must fall in the range 1 to 2." To resolve the error, update the fault domain to 2 and update `Sku` to `Aligned` as follows:

```
$avSet.PlatformFaultDomainCount = 2
Update-AzAvailabilitySet -AvailabilitySet $avSet -Sku Aligned
```

- Deallocate and migrate the VMs in the availability set. The following script deallocates each VM by using the [Stop-AzVM](#) cmdlet, converts it by using [ConvertTo-AzVMManagedDisk](#), and restarts it automatically as part of the migration process:

```
$avSet = Get-AzAvailabilitySet -ResourceGroupName $rgName -Name $avSetName

foreach($vmInfo in $avSet.VirtualMachinesReferences)
{
    $vm = Get-AzVM -ResourceGroupName $rgName | Where-Object {$_.Id -eq $vmInfo.id}
    Stop-AzVM -ResourceGroupName $rgName -Name $vm.Name -Force
    ConvertTo-AzVMManagedDisk -ResourceGroupName $rgName -VMName $vm.Name
}
```

## Troubleshooting

- Before converting, make sure all the VM extensions are in the 'Provisioning succeeded' state or the migration will fail with the error code 409.
- If there is an error during migration, or if a VM is in a failed state because of issues in a previous migration, run the `ConvertTo-AzVMManagedDisk` cmdlet again. A simple retry usually unblocks the situation.
- If you are converting a Linux VM to managed disks, use the latest version of the Azure Linux Agent. Operations using Azure Linux Agent versions '2.2.0' and earlier will likely fail. Running the migration on a generalized VM or a VM that belongs to a classic availability set is also not supported.
- If the migration fails with the "SnapshotCountExceeded" error, delete some snapshots and attempt the operation again.

## Migrate using the Azure portal

You can also migrate unmanaged disks to managed disks using the Azure portal.

1. Sign in to the [Azure portal](#).
2. Select the VM from the list of VMs in the portal.
3. In the blade for the VM, select **Disks** from the menu.
4. At the top of the **Disks** blade, select **Migrate to managed disks**.
5. If your VM is in an availability set, there will be a warning on the **Migrate to managed disks** blade that you need to migrate the availability set first. The warning should have a link you can click to migrate the availability set. Once the availability set is converted or if your VM is not in an availability set, click **Migrate** to start the process of migrating your disks to managed disks.

The VM will be stopped and restarted after migration is complete.

## Next steps

[Convert standard managed disks to premium](#)

Take a read-only copy of a VM by using [snapshots](#).

# Add a disk to a Linux VM

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to attach a persistent disk to your VM so that you can preserve your data - even if your VM is reprovisioned due to maintenance or resizing.

## Attach a new disk to a VM

If you want to add a new, empty data disk on your VM, use the `az vm disk attach` command with the `--new` parameter. If your VM is in an Availability Zone, the disk is automatically created in the same zone as the VM. For more information, see [Overview of Availability Zones](#). The following example creates a disk named *myDataDisk* that is 50 Gb in size:

```
az vm disk attach \
  -g myResourceGroup \
  --vm-name myVM \
  --name myDataDisk \
  --new \
  --size-gb 50
```

### Lower latency

In select regions, the disk attach latency has been reduced, so you'll see an improvement of up to 15%. This is useful if you have planned/unplanned failovers between VMs, you're scaling your workload, or are running a high scale stateful workload such as Azure Kubernetes Service. However, this improvement is limited to the explicit disk attach command, `az vm disk attach`. You won't see the performance improvement if you call a command that may implicitly perform an attach, like `az vm update`. You don't need to take any action other than calling the explicit attach command to see this improvement.

Lower latency is currently available in every public region except for:

- Canada Central
- Central US
- East US
- East US 2
- South Central US
- West US 2
- Germany North
- Jio India West
- North Europe
- West Europe

## Attach an existing disk

To attach an existing disk, find the disk ID and pass the ID to the `az vm disk attach` command. The following example queries for a disk named *myDataDisk* in *myResourceGroup*, then attaches it to the VM named *myVM*.

```
diskId=$(az disk show -g myResourceGroup -n myDataDisk --query 'id' -o tsv)
```

```
az vm disk attach -g myResourceGroup --vm-name myVM --name $diskId
```

## Format and mount the disk

To partition, format, and mount your new disk so your Linux VM can use it, SSH into your VM. For more information, see [How to use SSH with Linux on Azure](#). The following example connects to a VM with the public IP address of 10.123.123.25 with the username *azureuser*:

```
ssh azureuser@10.123.123.25
```

### Find the disk

Once connected to your VM, you need to find the disk. In this example, we are using `lsblk` to list the disks.

```
lsblk -o NAME,HCTL,SIZE,MOUNTPOINT | grep -i "sd"
```

The output is similar to the following example:

```
sda      0:0:0:0      30G
└─sda1        29.9G  /
└─sda14        4M
└─sda15       106M /boot/efi
sdb      1:0:1:0      14G
└─sdb1        14G /mnt
sdc      3:0:0:0      50G
```

Here, `sdc` is the disk that we want, because it is 50G. If you add multiple disks, and aren't sure which disk it is based on size alone, you can go to the VM page in the portal, select **Disks**, and check the LUN number for the disk under **Data disks**. Compare the LUN number from the portal to the last number of the HCTL portion of the output, which is the LUN.

### Format the disk

Format the disk with `parted`, if the disk size is 2 tebibytes (TiB) or larger then you must use GPT partitioning, if it is under 2TiB, then you can use either MBR or GPT partitioning.

#### NOTE

It is recommended that you use the latest version `parted` that is available for your distro. If the disk size is 2 tebibytes (TiB) or larger, you must use GPT partitioning. If disk size is under 2 TiB, then you can use either MBR or GPT partitioning.

The following example uses `parted` on `/dev/sdc`, which is where the first data disk will typically be on most VMs. Replace `sdc` with the correct option for your disk. We are also formatting it using the [XFS](#) filesystem.

```
sudo parted /dev/sdc --script mklabel gpt mkpart xfspart xfs 0% 100%
sudo mkfs.xfs /dev/sdc1
sudo partprobe /dev/sdc1
```

Use the `partprobe` utility to make sure the kernel is aware of the new partition and filesystem. Failure to use `partprobe` can cause the `blkid` or `lsblk` commands to not return the UUID for the new filesystem immediately.

### Mount the disk

Now, create a directory to mount the file system using `mkdir`. The following example creates a directory at `/datadrive`:

```
sudo mkdir /datadrive
```

Use `mount` to then mount the filesystem. The following example mounts the `/dev/sdc1` partition to the `/datadrive` mount point:

```
sudo mount /dev/sdc1 /datadrive
```

## Persist the mount

To ensure that the drive is remounted automatically after a reboot, it must be added to the `/etc/fstab` file. It is also highly recommended that the UUID (Universally Unique Identifier) is used in `/etc/fstab` to refer to the drive rather than just the device name (such as, `/dev/sdc1`). If the OS detects a disk error during boot, using the UUID avoids the incorrect disk being mounted to a given location. Remaining data disks would then be assigned those same device IDs. To find the UUID of the new drive, use the `blkid` utility:

```
sudo blkid
```

The output looks similar to the following example:

```
/dev/sda1: LABEL="cloudimg-rootfs" UUID="11111111-1b1b-1c1c-1d1d-1e1e1e1e1e1e" TYPE="ext4"
PARTUUID="1a1b1c1d-11aa-1234-1a1a1a1a1a1a"
/dev/sda15: LABEL="UEFI" UUID="BCD7-96A6" TYPE="vfat" PARTUUID="1e1g1cg1h-11aa-1234-1u1u1a1a1u1u"
/dev/sdb1: UUID="2222222-2b2b-2c2c-2d2d-2e2e2e2e2e" TYPE="ext4" TYPE="ext4" PARTUUID="1a2b3c4d-01"
/dev/sda14: PARTUUID="2e2g2cg2h-11aa-1234-1u1u1a1a1u1u"
/dev/sdc1: UUID="33333333-3b3b-3c3c-3d3d-3e3e3e3e3e" TYPE="xfs" PARTLABEL="xfspart" PARTUUID="c1c2c3c4-
1234-cdef-asdf3456ghjk"
```

### NOTE

Improperly editing the `/etc/fstab` file could result in an unbootable system. If unsure, refer to the distribution's documentation for information on how to properly edit this file. It is also recommended that a backup of the `/etc/fstab` file is created before editing.

Next, open the `/etc/fstab` file in a text editor as follows:

```
sudo nano /etc/fstab
```

In this example, use the UUID value for the `/dev/sdc1` device that was created in the previous steps, and the mountpoint of `/datadrive`. Add the following line to the end of the `/etc/fstab` file:

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e   /datadrive   xfs   defaults,nofail   1   2
```

In this example, we are using the nano editor, so when you are done editing the file, use `Ctrl+O` to write the file and `Ctrl+X` to exit the editor.

## NOTE

Later removing a data disk without editing fstab could cause the VM to fail to boot. Most distributions provide either the *nofail* and/or *nobootwait* fstab options. These options allow a system to boot even if the disk fails to mount at boot time. Consult your distribution's documentation for more information on these parameters.

The *nofail* option ensures that the VM starts even if the filesystem is corrupt or the disk does not exist at boot time. Without this option, you may encounter behavior as described in [Cannot SSH to Linux VM due to FSTAB errors](#).

The Azure VM Serial Console can be used for console access to your VM if modifying fstab has resulted in a boot failure. More details are available in the [Serial Console documentation](#).

## TRIM/UNMAP support for Linux in Azure

Some Linux kernels support TRIM/UNMAP operations to discard unused blocks on the disk. This feature is primarily useful in standard storage to inform Azure that deleted pages are no longer valid and can be discarded, and can save money if you create large files and then delete them.

There are two ways to enable TRIM support in your Linux VM. As usual, consult your distribution for the recommended approach:

- Use the `discard` mount option in `/etc/fstab`, for example:

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e3e /datadrive xfs defaults,discard 1 2
```

- In some cases, the `discard` option may have performance implications. Alternatively, you can run the `fstrim` command manually from the command line, or add it to your crontab to run regularly:

### Ubuntu

```
sudo apt-get install util-linux
sudo fstrim /datadrive
```

### RHEL/CentOS

```
sudo yum install util-linux
sudo fstrim /datadrive
```

## Troubleshooting

When adding data disks to a Linux VM, you may encounter errors if a disk does not exist at LUN 0. If you are adding a disk manually using the `az vm disk attach -new` command and you specify a LUN (`--lun`) rather than allowing the Azure platform to determine the appropriate LUN, take care that a disk already exists / will exist at LUN 0.

Consider the following example showing a snippet of the output from `lsscsi`:

```
[5:0:0:0]    disk    Msft    Virtual Disk    1.0    /dev/sdc
[5:0:0:1]    disk    Msft    Virtual Disk    1.0    /dev/sdd
```

The two data disks exist at LUN 0 and LUN 1 (the first column in the `lsscsi` output details `[host:channel:target:lun]`). Both disks should be accessible from within the VM. If you had manually specified the first disk to be added at LUN 1 and the second disk at LUN 2, you may not see the disks correctly from within your VM.

**NOTE**

The Azure `host` value is 5 in these examples, but this may vary depending on the type of storage you select.

This disk behavior is not an Azure problem, but the way in which the Linux kernel follows the SCSI specifications. When the Linux kernel scans the SCSI bus for attached devices, a device must be found at LUN 0 in order for the system to continue scanning for additional devices. As such:

- Review the output of `lsscsi` after adding a data disk to verify that you have a disk at LUN 0.
- If your disk does not show up correctly within your VM, verify a disk exists at LUN 0.

## Next steps

- To ensure your Linux VM is configured correctly, review the [Optimize your Linux machine performance](#) recommendations.
- Expand your storage capacity by adding additional disks and [configure RAID](#) for additional performance.

# Use the portal to attach a data disk to a Linux VM

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to attach both new and existing disks to a Linux virtual machine through the Azure portal. You can also [attach a data disk to a Windows VM in the Azure portal](#).

Before you attach disks to your VM, review these tips:

- The size of the virtual machine controls how many data disks you can attach. For details, see [Sizes for virtual machines](#).
- Disks attached to virtual machines are actually .vhdx files stored in Azure. For details, see our [Introduction to managed disks](#).
- After attaching the disk, you need to [connect to the Linux VM to mount the new disk](#).

## Find the virtual machine

1. Go to the [Azure portal](#) to find the VM. Search for and select **Virtual machines**.
2. Choose the VM from the list.
3. In the **Virtual machines** page, under **Settings**, choose **Disks**.

## Attach a new disk

1. On the **Disks** pane, under **Data disks**, select **Create and attach a new disk**.
2. Enter a name for your managed disk. Review the default settings, and update the **Storage type**, **Size (GiB)**, **Encryption** and **Host caching** as necessary.

LUN	Disk name	Storage type	Size (GiB)	Max IOPS	Max throughput (GiB/s)	Encryption	Host caching
0	Data disk name	Premium SSD	4	120	25	Platform-managed	None

3. When you are done, select **Save** at the top of the page to create the managed disk and update the VM configuration.

## Attach an existing disk

1. On the **Disks** pane, under **Data disks**, select **Attach existing disks**.
2. Click the drop-down menu for **Disk name** and select a disk from the list of available managed disks.
3. Click **Save** to attach the existing managed disk and update the VM configuration:

## Connect to the Linux VM to mount the new disk

To partition, format, and mount your new disk so your Linux VM can use it, SSH into your VM. For more information, see [How to use SSH with Linux on Azure](#). The following example connects to a VM with the public IP address of `10.123.123.25` with the username `azureuser`.

```
ssh azureuser@10.123.123.25
```

## Find the disk

Once connected to your VM, you need to find the disk. In this example, we are using `lsblk` to list the disks.

```
lsblk -o NAME,HCTL,SIZE,MOUNTPOINT | grep -i "sd"
```

The output is similar to the following example:

```
sda      0:0:0:0      30G
└─sda1        29.9G /
└─sda14       4M
└─sda15      106M /boot/efi
sdb      1:0:1:0      14G
└─sdb1        14G /mnt
sdc      3:0:0:0      4G
```

In this example, the disk that I added is `sdc`. It is a LUN 0 and is 4GB.

For a more complex example, here is what multiple data disks look like in the portal:

LUN	Disk name	Storage type	Size (GiB)
0	datadisk1	Premium SSD	4
1	datadisk2	Premium SSD	16
2	datadisk3	Standard HDD	32

In the image, you can see that there are 3 data disks: 4 GB on LUN 0, 16GB at LUN 1, and 32G at LUN 2.

Here is what that might look like using `lsblk`:

```
sda      0:0:0:0      30G
└─sda1        29.9G /
└─sda14       4M
└─sda15      106M /boot/efi
sdb      1:0:1:0      14G
└─sdb1        14G /mnt
sdc      3:0:0:0      4G
sdd      3:0:0:1      16G
sde      3:0:0:2      32G
```

From the output of `lsblk` you can see that the 4GB disk at LUN 0 is `sdc`, the 16GB disk at LUN 1 is `sdd`, and the 32G disk at LUN 2 is `sde`.

## Prepare a new empty disk

### IMPORTANT

If you are using an existing disk that contains data, skip to [mounting the disk](#). The following instructions will delete data on the disk.

If you are attaching a new disk, you need to partition the disk.

The `parted` utility can be used to partition and to format a data disk.

- It is recommended that you use the latest version `parted` that is available for your distro.
- If the disk size is 2 tebibytes (TiB) or larger, you must use GPT partitioning. If disk size is under 2 TiB, then you can use either MBR or GPT partitioning.

The following example uses `parted` on `/dev/sdc`, which is where the first data disk will typically be on most VMs. Replace `sdc` with the correct option for your disk. We are also formatting it using the [XFS](#) filesystem.

```
sudo parted /dev/sdc --script mklabel gpt mkpart xfspart xfs 0% 100%
sudo mkfs.xfs /dev/sdc1
sudo partprobe /dev/sdc1
```

Use the `partprobe` utility to make sure the kernel is aware of the new partition and filesystem. Failure to use `partprobe` can cause the `blkid` or `lsblk` commands to not return the UUID for the new filesystem immediately.

## Mount the disk

Create a directory to mount the file system using `mkdir`. The following example creates a directory at `/datadrive`:

```
sudo mkdir /datadrive
```

Use `mount` to then mount the filesystem. The following example mounts the `/dev/sdc1` partition to the `/datadrive` mount point:

```
sudo mount /dev/sdc1 /datadrive
```

To ensure that the drive is remounted automatically after a reboot, it must be added to the `/etc/fstab` file. It is also highly recommended that the UUID (Universally Unique Identifier) is used in `/etc/fstab` to refer to the drive rather than just the device name (such as, `/dev/sdc1`). If the OS detects a disk error during boot, using the UUID avoids the incorrect disk being mounted to a given location. Remaining data disks would then be assigned those same device IDs. To find the UUID of the new drive, use the `blkid` utility:

```
sudo blkid
```

The output looks similar to the following example:

```
/dev/sda1: LABEL="cloudimg-rootfs" UUID="11111111-1b1b-1c1c-1d1d-1e1e1e1e1e" TYPE="ext4"
PARTUUID="1a1b1c1d-11aa-1234-1a1a1a1a1a1a"
/dev/sda15: LABEL="UEFI" UUID="BCD7-96A6" TYPE="vfat" PARTUUID="1e1g1cg1h-11aa-1234-1u1u1a1a1u1u"
/dev/sdb1: UUID="2222222-2b2b-2c2c-2d2d-2e2e2e2e2e" TYPE="ext4" PARTUUID="1a2b3c4d-01"
/dev/sda14: PARTUUID="2e2g2cg2h-11aa-1234-1u1u1a1a1u1u"
/dev/sdc1: UUID="3333333-3b3b-3c3c-3d3d-3e3e3e3e3e" TYPE="xfs" PARTLABEL="xfspart" PARTUUID="c1c2c3c4-
1234-cdef-asdf3456ghjk"
```

### NOTE

Improperly editing the `/etc/fstab` file could result in an unbootable system. If unsure, refer to the distribution's documentation for information on how to properly edit this file. It is also recommended that a backup of the `/etc/fstab` file is created before editing.

Next, open the `/etc/fstab` file in a text editor as follows:

```
sudo nano /etc/fstab
```

In this example, use the UUID value for the `/dev/sdc1` device that was created in the previous steps, and the mountpoint of `/datadrive`. Add the following line to the end of the `/etc/fstab` file:

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e /datadrive xfs defaults,nofail 1 2
```

We used the nano editor, so when you are done editing the file, use `Ctrl+O` to write the file and `Ctrl+X` to exit the editor.

#### NOTE

Later removing a data disk without editing fstab could cause the VM to fail to boot. Most distributions provide either the `nofail` and/or `nobootwait` fstab options. These options allow a system to boot even if the disk fails to mount at boot time. Consult your distribution's documentation for more information on these parameters.

The `nofail` option ensures that the VM starts even if the filesystem is corrupt or the disk does not exist at boot time. Without this option, you may encounter behavior as described in [Cannot SSH to Linux VM due to FSTAB errors](#)

## Verify the disk

You can now use `lsblk` again to see the disk and the mountpoint.

```
lsblk -o NAME,HCTL,SIZE,MOUNTPOINT | grep -i "sd"
```

The output will look something like this:

```
sda      0:0:0:0      30G
└─sda1        29.9G /
└─sda14       4M
└─sda15     106M /boot/efi
sdb      1:0:1:0      14G
└─sdb1        14G /mnt
sdc      3:0:0:0      4G
└─sdc1        4G /datadrive
```

You can see that `sdc` is now mounted at `/datadrive`.

### TRIM/UNMAP support for Linux in Azure

Some Linux kernels support TRIM/UNMAP operations to discard unused blocks on the disk. This feature is primarily useful in standard storage to inform Azure that deleted pages are no longer valid and can be discarded, and can save money if you create large files and then delete them.

There are two ways to enable TRIM support in your Linux VM. As usual, consult your distribution for the recommended approach:

- Use the `discard` mount option in `/etc/fstab`, for example:

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e /datadrive xfs defaults,discard 1 2
```

- In some cases, the `discard` option may have performance implications. Alternatively, you can run the

`fstrim` command manually from the command line, or add it to your crontab to run regularly:

## Ubuntu

```
sudo apt-get install util-linux  
sudo fstrim /datadrive
```

## RHEL/CentOS

```
sudo yum install util-linux  
sudo fstrim /datadrive
```

## Next steps

For more information, and to help troubleshoot disk issues, see [Troubleshoot Linux VM device name changes](#).

You can also [attach a data disk](#) using the Azure CLI.

# Attach a data disk to a Windows VM with PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

This article shows you how to attach both new and existing disks to a Windows virtual machine by using PowerShell.

First, review these tips:

- The size of the virtual machine controls how many data disks you can attach. For more information, see [Sizes for virtual machines](#).
- To use premium SSDs, you'll need a [premium storage-enabled VM type](#), like the DS-series or GS-series virtual machine.

This article uses PowerShell within the [Azure Cloud Shell](#), which is constantly updated to the latest version. To open the Cloud Shell, select **Try it** from the top of any code block.

## Lower latency

In select regions, the disk attach latency has been reduced, so you'll see an improvement of up to 15%. This is useful if you have planned/unplanned failovers between VMs, you're scaling your workload, or are running a high scale stateful workload such as Azure Kubernetes Service. However, this improvement is limited to the explicit disk attach command, `Add-AzVMDataDisk`. You won't see the performance improvement if you call a command that may implicitly perform an attach, like `Update-AzVM`. You don't need to take any action other than calling the explicit attach command to see this improvement.

Lower latency is currently available in every public region except for:

- Canada Central
- Central US
- East US
- East US 2
- South Central US
- West US 2
- Germany North
- Jio India West
- North Europe
- West Europe

## Add an empty data disk to a virtual machine

This example shows how to add an empty data disk to an existing virtual machine.

### Using managed disks

```

$rgName = 'myResourceGroup'
$vmName = 'myVM'
$location = 'East US'
$storageType = 'Premium_LRS'
$dataDiskName = $vmName + '_datadisk1'

$diskConfig = New-AzDiskConfig -SkuName $storageType -Location $location -CreateOption Empty -DiskSizeGB 128
$dataDisk1 = New-AzDisk -DiskName $dataDiskName -Disk $diskConfig -ResourceGroupName $rgName

$vm = Get-AzVM -Name $vmName -ResourceGroupName $rgName
$vm = Add-AzVMDataDisk -VM $vm -Name $dataDiskName -CreateOption Attach -ManagedDiskId $dataDisk1.Id -Lun 1

Update-AzVM -VM $vm -ResourceGroupName $rgName

```

## Using managed disks in an Availability Zone

To create a disk in an Availability Zone, use [New-AzDiskConfig](#) with the `-Zone` parameter. The following example creates a disk in zone 1.

```

$rgName = 'myResourceGroup'
$vmName = 'myVM'
$location = 'East US 2'
$storageType = 'Premium_LRS'
$dataDiskName = $vmName + '_datadisk1'

$diskConfig = New-AzDiskConfig -SkuName $storageType -Location $location -CreateOption Empty -DiskSizeGB 128
-Zone 1
$dataDisk1 = New-AzDisk -DiskName $dataDiskName -Disk $diskConfig -ResourceGroupName $rgName

$vm = Get-AzVM -Name $vmName -ResourceGroupName $rgName
$vm = Add-AzVMDataDisk -VM $vm -Name $dataDiskName -CreateOption Attach -ManagedDiskId $dataDisk1.Id -Lun 1

Update-AzVM -VM $vm -ResourceGroupName $rgName

```

## Initialize the disk

After you add an empty disk, you'll need to initialize it. To initialize the disk, you can sign in to a VM and use disk management. If you enabled [WinRM](#) and a certificate on the VM when you created it, you can use remote PowerShell to initialize the disk. You can also use a custom script extension:

```

$location = "location-name"
$scriptName = "script-name"
$fileName = "script-file-name"
Set-AzVMCustomScriptExtension -ResourceGroupName $rgName -Location $locName -VMName $vmName -Name
$scriptName -TypeHandlerVersion "1.4" -StorageAccountName "mystore1" -StorageAccountKey "primary-key" -
FileName $fileName -ContainerName "scripts"

```

The script file can contain code to initialize the disks, for example:

```
$disks = Get-Disk | Where partitionstyle -eq 'raw' | sort number

$letters = 70..89 | ForEach-Object { [char]$_ }

$count = 0
$labels = "data1","data2"

foreach ($disk in $disks) {
    $driveLetter = $letters[$count].ToString()
    $disk |
        Initialize-Disk -PartitionStyle MBR -PassThru |
        New-Partition -UseMaximumSize -DriveLetter $driveLetter |
        Format-Volume -FileSystem NTFS -NewFileSystemLabel $labels[$count] -Confirm:$false -Force
$count++
}
```

## Attach an existing data disk to a VM

You can attach an existing managed disk to a VM as a data disk.

```
$rgName = "myResourceGroup"
$vmName = "myVM"
$dataDiskName = "myDisk"
$disk = Get-AzDisk -ResourceGroupName $rgName -DiskName $dataDiskName

$vm = Get-AzVM -Name $vmName -ResourceGroupName $rgName

$vm = Add-AzVMDataDisk -CreateOption Attach -Lun 0 -VM $vm -ManagedDiskId $disk.Id

Update-AzVM -VM $vm -ResourceGroupName $rgName
```

## Next steps

You can also deploy managed disks using templates. For more information, see [Using Managed Disks in Azure Resource Manager Templates](#) or the [quickstart template](#) for deploying multiple data disks.

# Attach a managed data disk to a Windows VM by using the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

This article shows you how to attach a new managed data disk to a Windows virtual machine (VM) by using the Azure portal. The size of the VM determines how many data disks you can attach. For more information, see [Sizes for virtual machines](#).

## Add a data disk

1. Sign in to the [Azure portal](#).
2. Search for and select **Virtual machines**.
3. Select a virtual machine from the list.
4. On the **Virtual machine** pane, select **Disks**.
5. On the **Disks** pane, select **Create and attach a new disk**.
6. In the drop-downs for the new disk, make the selections you want, and name the disk.
7. Select **Save** to create and attach the new data disk to the VM.

## Initialize a new data disk

1. Connect to the VM.
2. Select the Windows **Start** menu inside the running VM and enter **diskmgmt.msc** in the search box. The **Disk Management** console opens.
3. Disk Management recognizes that you have a new, uninitialized disk and the **Initialize Disk** window appears.
4. Verify the new disk is selected and then select **OK** to initialize it.
5. The new disk appears as **unallocated**. Right-click anywhere on the disk and select **New simple volume**. The **New Simple Volume Wizard** window opens.
6. Proceed through the wizard, keeping all of the defaults, and when you're done select **Finish**.
7. Close **Disk Management**.
8. A pop-up window appears notifying you that you need to format the new disk before you can use it. Select **Format disk**.
9. In the **Format new disk** window, check the settings, and then select **Start**.
10. A warning appears notifying you that formatting the disks erases all of the data. Select **OK**.
11. When the formatting is complete, select **OK**.

## Next steps

- You can also [attach a data disk by using PowerShell](#).
- If your application needs to use the *D:* drive to store data, you can [change the drive letter of the Windows temporary disk](#).

# Use the D: drive as a data drive on a Windows VM

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

If your application needs to use the D drive to store data, follow these instructions to use a different drive letter for the temporary disk. Never use the temporary disk to store data that you need to keep.

If you resize or **Stop (Deallocate)** a virtual machine, this may trigger placement of the virtual machine to a new hypervisor. A planned or unplanned maintenance event may also trigger this placement. In this scenario, the temporary disk will be reassigned to the first available drive letter. If you have an application that specifically requires the D: drive, you need to follow these steps to temporarily move the pagefile.sys, attach a new data disk and assign it the letter D and then move the pagefile.sys back to the temporary drive. Once complete, Azure will not take back the D: if the VM moves to a different hypervisor.

For more information about how Azure uses the temporary disk, see [Understanding the temporary drive on Microsoft Azure Virtual Machines](#)

## Attach the data disk

First, you'll need to attach the data disk to the virtual machine. To do this using the portal, see [How to attach a managed data disk in the Azure portal](#).

## Temporarily move pagefile.sys to C drive

1. Connect to the virtual machine.
2. Right-click the **Start** menu and select **System**.
3. In the left-hand menu, search for and select **View advanced system settings**.
4. In the **Performance** section, select **Settings**.
5. Select the **Advanced** tab.
6. In the **Virtual memory** section, select **Change**.
7. Select the C drive and then click **System managed size** and then click **Set**.
8. Select the D drive and then click **No paging file** and then click **Set**.
9. Click **Apply**. You will get a warning that the computer needs to be restarted for the changes to take affect.
10. Restart the virtual machine.

## Change the drive letters

1. Once the VM restarts, log back on to the VM.
2. Click the **Start** menu and type **diskmgmt.msc** and hit Enter. Disk Management will start.
3. Right-click on D, the Temporary Storage drive, and select **Change Drive Letter and Paths**.
4. Under Drive letter, select a new drive such as T and then click **OK**.
5. Right-click on the data disk, and select **Change Drive Letter and Paths**.
6. Under Drive letter, select drive D and then click **OK**.

## Move pagefile.sys back to the temporary storage drive

1. Right-click the **Start** menu and select **System**
2. In the left-hand menu, search for and select **View advanced system settings**.

3. In the **Performance** section, select **Settings**.
4. Select the **Advanced** tab.
5. In the **Virtual memory** section, select **Change**.
6. Select the OS drive C and click **No paging file** and then click **Set**.
7. Select the temporary storage drive T and then click **System managed size** and then click **Set**.
8. Click **Apply**. You will get a warning that the computer needs to be restarted for the changes to take affect.
9. Restart the virtual machine.

## Next steps

- You can increase the storage available to your virtual machine by [attaching an additional data disk](#).

# How to detach a data disk from a Linux virtual machine

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you no longer need a data disk that's attached to a virtual machine, you can easily detach it. This removes the disk from the virtual machine, but doesn't remove it from storage. In this article, we are working with an Ubuntu LTS 16.04 distribution. If you are using a different distribution, the instructions for unmounting the disk might be different.

## WARNING

If you detach a disk it is not automatically deleted. If you have subscribed to Premium storage, you will continue to incur storage charges for the disk. For more information, see [Pricing and Billing when using Premium Storage](#).

If you want to use the existing data on the disk again, you can reattach it to the same virtual machine, or another one.

## Connect to the VM to unmount the disk

Before you can detach the disk using either CLI or the portal, you need to unmount the disk and removed references to it from your fstab file.

Connect to the VM. In this example, the public IP address of the VM is *10.0.1.4* with the username *azureuser*:

```
ssh azureuser@10.0.1.4
```

First, find the data disk that you want to detach. The following example uses dmesg to filter on SCSI disks:

```
dmesg | grep SCSI
```

The output is similar to the following example:

```
[    0.294784] SCSI subsystem initialized
[    0.573458] Block layer SCSI generic (bsg) driver version 0.4 loaded (major 252)
[    7.110271] sd 2:0:0:0: [sda] Attached SCSI disk
[   8.079653] sd 3:0:1:0: [sdb] Attached SCSI disk
[ 1828.162306] sd 5:0:0:0: [sdc] Attached SCSI disk
```

Here, *sdc* is the disk that we want to detach. You also should grab the UUID of the disk.

```
sudo -i blkid
```

The output looks similar to the following example:

```
/dev/sda1: UUID="11111111-1b1b-1c1c-1d1d-1e1e1e1e1e" TYPE="ext4"
/dev/sdb1: UUID="22222222-2b2b-2c2c-2d2d-2e2e2e2e2e" TYPE="ext4"
/dev/sdc1: UUID="33333333-3b3b-3c3c-3d3d-3e3e3e3e3e" TYPE="ext4"
```

Edit the `/etc/fstab` file to remove references to the disk.

#### NOTE

Improperly editing the `/etc/fstab` file could result in an unbootable system. If unsure, refer to the distribution's documentation for information on how to properly edit this file. It is also recommended that a backup of the `/etc/fstab` file is created before editing.

Open the `/etc/fstab` file in a text editor as follows:

```
sudo vi /etc/fstab
```

In this example, the following line needs to be deleted from the `/etc/fstab` file:

```
UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e    /datadrive    ext4    defaults,nofail    1    2
```

Use `umount` to unmount the disk. The following example unmounts the `/dev/sdc1` partition from the `/datadrive` mount point:

```
sudo umount /dev/sdc1 /datadrive
```

## Detach a data disk using Azure CLI

This example detaches the `myDataDisk` disk from VM named `myVM` in `myResourceGroup`.

```
az vm disk detach \
  -g myResourceGroup \
  --vm-name myVm \
  -n myDataDisk
```

The disk stays in storage but is no longer attached to a virtual machine.

#### Lower latency

In select regions, the disk detach latency has been reduced, so you'll see an improvement of up to 15%. This is useful if you have planned/unplanned failovers between VMs, you're scaling your workload, or are running a high scale stateful workload such as Azure Kubernetes Service. However, this improvement is limited to the explicit disk detach command, `az vm disk detach`. You won't see the performance improvement if you call a command that may implicitly perform a detach, like `az vm update`. You don't need to take any action other than calling the explicit detach command to see this improvement.

Lower latency is currently available in every public region except for:

- Canada Central
- Central US
- East US
- East US 2
- South Central US

- West US 2
- Germany North
- Jio India West
- North Europe
- West Europe

## Detach a data disk using the portal

1. In the left menu, select **Virtual Machines**.
2. In the virtual machine blade, select **Disks**.
3. In the **Disks** blade, to the far right of the data disk that you would like to detach, select the **X** button, to detach the disk.
4. After the disk has been removed, select **Save** on the top of the blade.

The disk stays in storage but is no longer attached to a virtual machine. The disk is not deleted.

## Next steps

If you want to reuse the data disk, you can just [attach it to another VM](#).

If you want to delete the disk, so that you no longer incur storage costs, see [Find and delete unattached Azure managed and unmanaged disks - Azure portal](#).

# How to detach a data disk from a Windows virtual machine

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

When you no longer need a data disk that's attached to a virtual machine, you can easily detach it. This removes the disk from the virtual machine, but doesn't remove it from storage.

## WARNING

If you detach a disk it is not automatically deleted. If you have subscribed to Premium storage, you will continue to incur storage charges for the disk. For more information, see [Pricing and Billing when using Premium Storage](#).

If you want to use the existing data on the disk again, you can reattach it to the same virtual machine, or another one.

## Detach a data disk using PowerShell

You can *hot* remove a data disk using PowerShell, but make sure nothing is actively using the disk before detaching it from the VM.

In this example, we remove the disk named `myDisk` from the VM `myVM` in the `myResourceGroup` resource group. First you remove the disk using the `Remove-AzVMDataDisk` cmdlet. Then, you update the state of the virtual machine, using the `Update-AzVM` cmdlet, to complete the process of removing the data disk.

```
$VirtualMachine = Get-AzVM ` 
    -ResourceGroupName "myResourceGroup" ` 
    -Name "myVM"
Remove-AzVMDataDisk ` 
    -VM $VirtualMachine ` 
    -Name "myDisk"
Update-AzVM ` 
    -ResourceGroupName "myResourceGroup" ` 
    -VM $VirtualMachine
```

The disk stays in storage but is no longer attached to a virtual machine.

## Lower latency

In select regions, the disk detach latency has been reduced, so you'll see an improvement of up to 15%. This is useful if you have planned/unplanned failovers between VMs, you're scaling your workload, or are running a high scale stateful workload such as Azure Kubernetes Service. However, this improvement is limited to the explicit disk detach command, `Remove-AzVMDataDisk`. You won't see the performance improvement if you call a command that may implicitly perform a detach, like `Update-AzVM`. You don't need to take any action other than calling the explicit detach command to see this improvement.

Lower latency is currently available in every public region except for:

- Canada Central
- Central US
- East US

- East US 2
- South Central US
- West US 2
- Germany North
- Jio India West
- North Europe
- West Europe

## Detach a data disk using the portal

You can *hot* remove a data disk, but make sure nothing is actively using the disk before detaching it from the VM.

1. In the left menu, select **Virtual Machines**.
2. Select the virtual machine that has the data disk you want to detach.
3. Under **Settings**, select **Disks**.
4. In the **Disks** pane, to the far right of the data disk that you would like to detach, select the X button to detach.
5. Select **Save** on the top of the page to save your changes.

The disk stays in storage but is no longer attached to a virtual machine. The disk isn't deleted.

## Next steps

If you want to reuse the data disk, you can just [attach it to another VM](#).

If you want to delete the disk, so that you no longer incur storage costs, see [Find and delete unattached Azure managed and unmanaged disks - Azure portal](#).

# Expand virtual hard disks on a Linux VM with the Azure CLI

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article describes how to expand managed disks for a Linux virtual machine (VM) with the Azure CLI. You can [add data disks](#) to provide for additional storage space, and you can also expand an existing data disk. The default virtual hard disk size for the operating system (OS) is typically 30 GB on a Linux VM in Azure. This article covers expanding either OS disks or data disks.

## WARNING

Always make sure that your filesystem is in a healthy state, your disk partition table type will support the new size, and ensure your data is backed up before you perform disk expansion operations. For more information, see the [Azure Backup quickstart](#).

## Expand an Azure Managed Disk

### Expand without downtime

You can now expand your managed disks without deallocating your VM.

This feature has the following limitations:

- Only supported for data disks.
- If a disk is 4 TiB or less, you can't expand it beyond 4 TiB without deallocating the VM. If a disk is already greater than 4 TiB, you can expand it without deallocating the VM.
- Not supported for Ultra disks, Premium SSD v2 (preview) disks, or standard HDDs.
- Not supported for shared disks.
- Install and use either:
  - The [latest Azure CLI](#)
  - The [latest Azure PowerShell module](#)
  - The [Azure portal](#)
  - Or an Azure Resource Manager template with an API version that's 2021-04-01 or newer.

### Get started

Make sure that you have the latest [Azure CLI](#) installed and are signed in to an Azure account by using [az login](#).

This article requires an existing VM in Azure with at least one data disk attached and prepared. If you do not already have a VM that you can use, see [Create and prepare a VM with data disks](#).

In the following samples, replace example parameter names such as `myResourceGroup` and `myVM` with your own values.

## IMPORTANT

If your disk meets the requirements in [Expand without downtime](#), you can skip step 1 and 3.

- Operations on virtual hard disks can't be performed with the VM running. Deallocate your VM with `az vm deallocate`. The following example deallocates the VM named *myVM* in the resource group named *myResourceGroup*.

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

**NOTE**

The VM must be deallocated to expand the virtual hard disk. Stopping the VM with `az vm stop` does not release the compute resources. To release compute resources, use `az vm deallocate`.

- View a list of managed disks in a resource group with `az disk list`. The following example displays a list of managed disks in the resource group named *myResourceGroup*.

```
az disk list \  
  --resource-group myResourceGroup \  
  --query '[*].{Name:name,Gb:diskSizeGb,Tier:accountType}' \  
  --output table
```

Expand the required disk with `az disk update`. The following example expands the managed disk named *myDataDisk* to 200 GB:

```
az disk update \  
  --resource-group myResourceGroup \  
  --name myDataDisk \  
  --size-gb 200
```

**NOTE**

When you expand a managed disk, the updated size is rounded up to the nearest managed disk size. For a table of the available managed disk sizes and tiers, see [Azure Managed Disks Overview - Pricing and Billing](#).

- Start your VM with `az vm start`. The following example starts the VM named *myVM* in the resource group named *myResourceGroup*.

```
az vm start --resource-group myResourceGroup --name myVM
```

## Expand a disk partition and filesystem

To use an expanded disk, expand the underlying partition and filesystem.

- SSH to your VM with the appropriate credentials. You can see the public IP address of your VM with `az vm show`:

```
az vm show --resource-group myResourceGroup --name myVM -d --query [publicIps] --output tsv
```

- Expand the underlying partition and filesystem.

- If the disk is already mounted, unmount it:

```
sudo umount /dev/sdc1
```

b. Use `parted` to view disk information and resize the partition:

```
sudo parted /dev/sdc
```

View information about the existing partition layout with `print`. The output is similar to the following example, which shows the underlying disk is 215 GB:

```
GNU Parted 3.2
Using /dev/sdc1
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Model: Unknown Msft Virtual Disk (scsi)
Disk /dev/sdc1: 215GB
Sector size (logical/physical): 512B/4096B
Partition Table: loop
Disk Flags:

Number  Start   End     Size   File system  Flags
      0.00B  107GB  107GB   ext4
```

c. Expand the partition with `resizepart`. Enter the partition number, *1*, and a size for the new partition:

```
(parted) resizepart
Partition number? 1
End? [107GB]? 215GB
```

d. To exit, enter `quit`.

3. With the partition resized, verify the partition consistency with `e2fsck`:

```
sudo e2fsck -f /dev/sdc1
```

4. Resize the filesystem with `resize2fs`:

```
sudo resize2fs /dev/sdc1
```

5. Mount the partition to the desired location, such as `/datadrive`:

```
sudo mount /dev/sdc1 /datadrive
```

6. To verify the data disk has been resized, use `df -h`. The following example output shows the data drive `/dev/sdc1` is now 200 GB:

```
Filesystem      Size  Used  Avail Use% Mounted on
/dev/sdc1       197G  60M  187G   1% /datadrive
```

## Next steps

- If you need additional storage, you can also [add data disks to a Linux VM](#).

- For more information about disk encryption, see [Azure Disk Encryption for Linux VMs](#).

# How to expand virtual hard disks attached to a Windows virtual machine

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

When you create a new virtual machine (VM) in a resource group by deploying an image from [Azure Marketplace](#), the default operating system (OS) disk is usually 127 GiB (some images have smaller OS disk sizes by default). You can add data disks to your VM (the amount depends on the VM SKU you selected) and we recommend installing applications and CPU-intensive workloads on data disks. You may need to expand the OS disk if you're supporting a legacy application that installs components on the OS disk or if you're migrating a physical PC or VM from on-premises that has a larger OS disk. This article covers expanding either OS disks or data disks.

## IMPORTANT

Unless you use [Expand without downtime](#), expanding a data disk requires the VM to be deallocated.

Shrinking an existing disk isn't supported and may result in data loss.

After expanding the disks, you need to [Expand the volume in the operating system](#) to take advantage of the larger disk.

## Expand without downtime

You can now expand your data disks without deallocating your VM.

This feature has the following limitations:

- Only supported for data disks.
- If a disk is 4 TiB or less, you can't expand it beyond 4 TiB without deallocating the VM. If a disk is already greater than 4 TiB, you can expand it without deallocating the VM.
- Not supported for Ultra disks, Premium SSD v2 (preview) disks, or standard HDDs.
- Not supported for shared disks.
- Install and use either:
  - The [latest Azure CLI](#)
  - The [latest Azure PowerShell module](#)
  - The [Azure portal](#)
  - Or an Azure Resource Manager template with an API version that's 2021-04-01 or newer.

## Resize a managed disk in the Azure portal

## IMPORTANT

If your disk meets the requirements in [Expand without downtime](#), you can skip step 1.

1. In the [Azure portal](#), go to the virtual machine in which you want to expand the disk. Select **Stop** to deallocate the VM.
2. In the left menu under **Settings**, select **Disks**.

**Essentials**

Resource group (change) [reproRG](#)

Status Stopped (deallocated)

Location West US 2

Subscription (change) [Microsoft Azure Internal Consumption](#)

Subscription ID example-subscription-id-sequence

Tags (change) [Click here to add tags](#)

Operating system Windows

Size Standard D2s v3 (2 vcpus, 8 GiB memory)

Public IP address [windowsRepro-ip](#)

Virtual network/subnet [reproRG-vnet/default](#)

DNS name [Configure](#)

**Properties** Monitoring Capabilities Recommendations Tutorials

**Virtual machine** **Networking**

3. Under Disk name, select the disk you want to expand.

Disk name	Storage type	Size (GiB)
windowsRepro_disk1_1c8	Premium SSD	127

4. In the left menu under Settings, select Size + performance.

Show data for last: 1 hour 6 hours 12 hours 1 day 7 days

Series	Value
OS Disk Read Bytes/S... win10vm	401.9 kB/s
OS Disk Write Bytes/... win10vm	285.7 kB/s
OS Disk Read Operat... win10vm	8.97 /s
OS Disk Write Operat... win10vm	10.76 /s

5. In Size + performance, select the disk size you want.

**WARNING**

The new size should be greater than the existing disk size. The maximum allowed is 4,095 GB for OS disks. (It's possible to expand the VHD blob beyond that size, but the OS works only with the first 4,095 GB of space.)

Size	Disk tier	Provisioned IOPS	Provisioned through...	Max Shares	Max burst IOPS	Max burst throughput
4 GiB	P1	120	25	-	3500	170
8 GiB	P2	120	25	-	3500	170
16 GiB	P3	120	25	-	3500	170
32 GiB	P4	120	25	-	3500	170
64 GiB	P6	240	50	-	3500	170
128 GiB	P10	500	100	-	3500	170
256 GiB	P15	1100	125	2	3500	170
512 GiB	P20	2300	150	2	3500	170
1024 GiB	P30	5000	200	5	-	-
2048 GiB	P40	7500	250	5	-	-
4096 GiB	P50	7500	250	5	-	-
8192 GiB	P60	16000	500	10	-	-
16384 GiB	P70	18000	750	10	-	-
32767 GiB	P80	20000	900	10	-	-

Custom disk size (GiB) \*

**Resize** **Discard**

## 6. Select **Resize** at the bottom of the page.

Size	Disk tier	Provisioned IOPS	Provisioned through...	Max Shares	Max burst IOPS	Max burst throughput
4 GiB	P1	120	25	-	3500	170
8 GiB	P2	120	25	-	3500	170
16 GiB	P3	120	25	-	3500	170
32 GiB	P4	120	25	-	3500	170
64 GiB	P6	240	50	-	3500	170
128 GiB	P10	500	100	-	3500	170
256 GiB	P15	1100	125	2	3500	170
512 GiB	P20	2300	150	2	3500	170
1024 GiB	P30	5000	200	5	-	-
2048 GiB	P40	7500	250	5	-	-
4096 GiB	P50	7500	250	5	-	-
8192 GiB	P60	16000	500	10	-	-
16384 GiB	P70	18000	750	10	-	-
32767 GiB	P80	20000	900	10	-	-

Custom disk size (GiB) \*

**Resize** **Discard**

## Resize a managed disk by using PowerShell

Open your PowerShell ISE or PowerShell window in administrative mode and follow the steps below:

Sign in to your Microsoft Azure account in resource management mode and select your subscription:

```
Connect-AzAccount
Select-AzSubscription -SubscriptionName 'my-subscription-name'
```

Set your resource group name and VM name:

```
$rgName = 'my-resource-group-name'
$vmName = 'my-vm-name'
$diskName = 'my-disk-name'
```

Obtain a reference to your VM:

```
$vm = Get-AzVM -ResourceGroupName $rgName -Name $vmName
```

### IMPORTANT

If your disk meets the requirements in [expand without downtime](#), you can skip step 4 and 6.

Stop the VM before resizing the disk:

```
Stop-AzVM -ResourceGroupName $rgName -Name $vmName
```

Obtain a reference to the managed OS disk. Set the size of the managed OS disk to the desired value and update the Disk:

```
$disk= Get-AzDisk -ResourceGroupName $rgName -DiskName $diskName  
$disk.DiskSizeGB = 1023  
Update-AzDisk -ResourceGroupName $rgName -Disk $disk -DiskName $disk.Name
```

### WARNING

The new size should be greater than the existing disk size. The maximum allowed is 4,095 GB for OS disks. (It is possible to expand the VHD blob beyond that size, but the OS works only with the first 4,095 GB of space.)

Updating the VM might take a few seconds. When the command finishes executing, restart the VM:

```
Start-AzVM -ResourceGroupName $rgName -Name $vmName
```

Remote into the VM, open **Computer Management** (or **Disk Management**) and expand the drive using the newly allocated space.

## Expand the volume in the operating system

When you've expanded the disk for the VM, you need to go into the OS and expand the volume to encompass the new space. There are several methods for expanding a partition. This section covers connecting the VM using an RDP connection to expand the partition using [Using Diskpart](#) or [Using Disk Manager](#).

### Using DiskPart

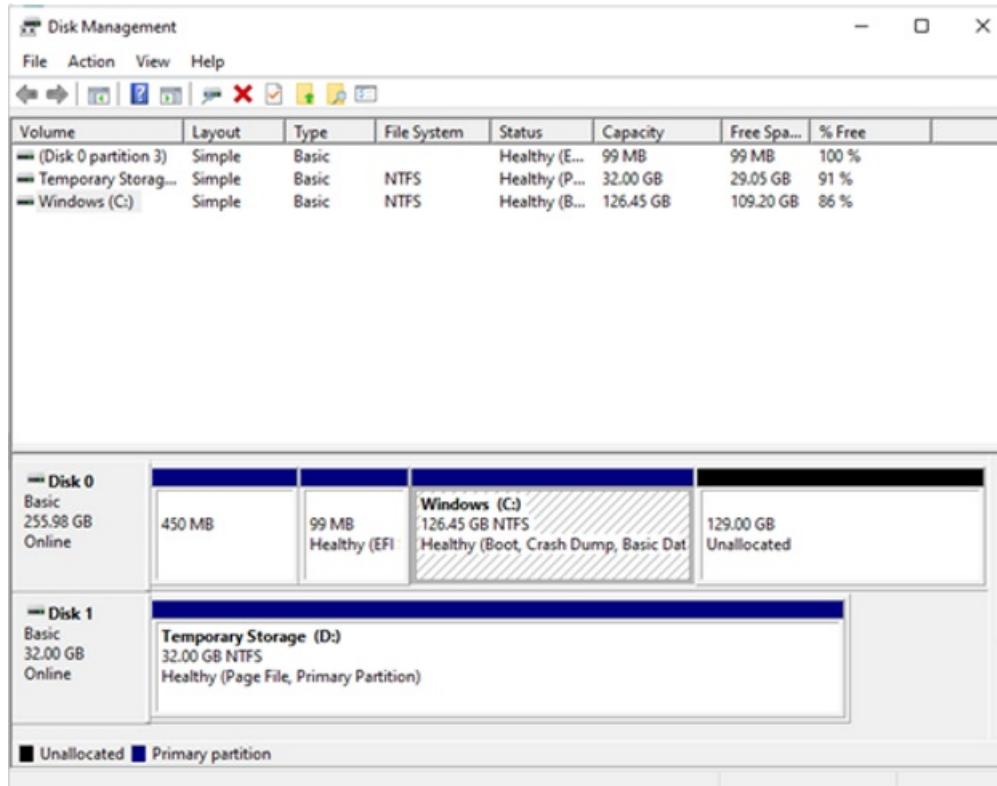
When you've expanded the disk for the VM, you need to go into the OS and expand the volume to encompass the new space. There are several methods for expanding a partition. This section covers connecting the VM using an RDP connection to expand the partition using **DiskPart**.

1. Open an RDP connection to your VM.
2. Open a command prompt and type **diskpart**.
3. At the **DISKPART** prompt, type `list volume`. Make note of the volume you want to extend.
4. At the **DISKPART** prompt, type `select volume <volumenumber>`. This selects the volume *volumenumber* that you want to extend into contiguous, empty space on the same disk.
5. At the **DISKPART** prompt, type `extend [size=<size>]`. This extends the selected volume by *size* in megabytes (MB).

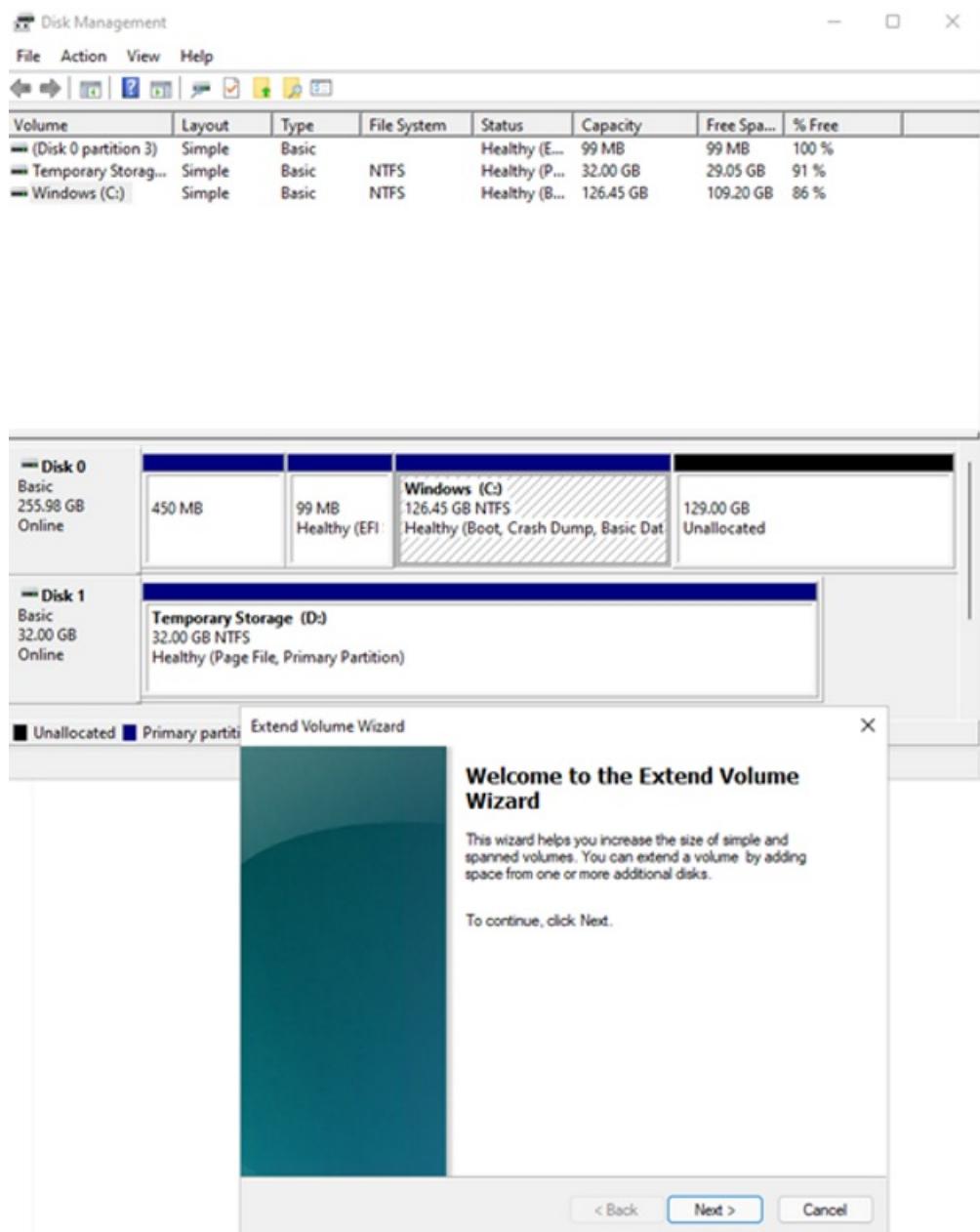
## Using Disk Manager

1. Start a remote desktop session with the VM.

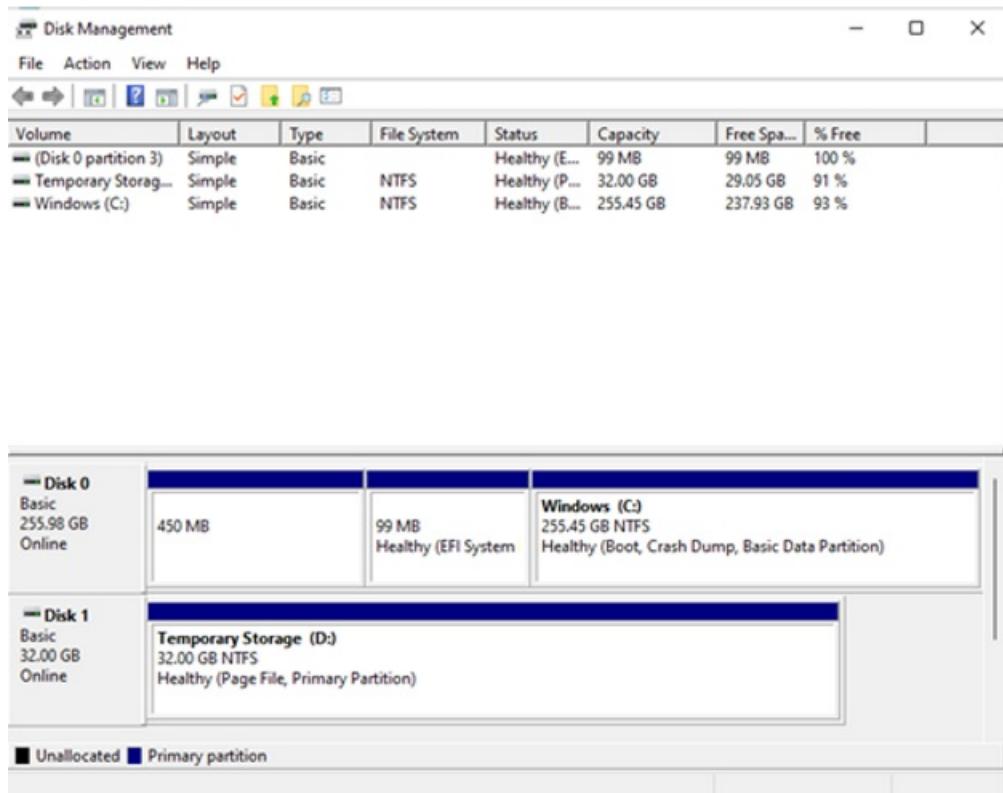
2. Open Disk Management.



3. Right-click on existing C: drive partition -> Extend Volume.



4. Follow the steps you should be able to see the disk with updated capacity:



## Next steps

You can also attach disks using the [Azure portal](#).

# Expand unmanaged virtual hard disks attached to a virtual machine

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article covers how to expand unmanaged disks. To learn how to expand a managed disk, use either the [Windows](#) or [Linux](#) articles.

**Applies to:** ✓ Windows VMs ✓ Linux VMs ✓ Flexible scale sets

When you create a new virtual machine (VM) in a resource group by deploying an image from [Azure Marketplace](#), the default operating system (OS) drive is often 127 GB (some images have smaller OS disk sizes by default). Even though it's possible to add data disks to the VM (the number depends on the SKU you chose) and we recommend installing applications and CPU-intensive workloads on these addendum disks, often, customers need to expand the OS drive to support specific scenarios:

- To support legacy applications that install components on the OS drive.
- To migrate a physical PC or VM from on-premises with a larger OS drive.

## IMPORTANT

Resizing an OS or data disk of an Azure VM requires the VM to be deallocated.

Shrinking an existing disk isn't supported, and can potentially result in data loss.

After expanding the disks, you need to expand the volume within the OS in either [Windows](#) or [Linux](#) to take advantage of the larger disk.

## Resize an unmanaged disk by using PowerShell

Open your PowerShell ISE or PowerShell window in administrative mode and follow the steps below:

- Sign in to your Microsoft Azure account in resource management mode and select your subscription:

```
Connect-AzAccount  
Select-AzSubscription -SubscriptionName 'my-subscription-name'
```

- Set your resource group name and VM names:

```
$rgName = 'my-resource-group-name'  
$vmName = 'my-vm-name'
```

- Obtain a reference to your VM:

```
$vm = Get-AzVM -ResourceGroupName $rgName -Name $vmName
```

- Stop the VM before resizing the disk:

```
Stop-AzVM -ResourceGroupName $rgName -Name $vmName
```

- Set the size of the unmanaged OS disk to the desired value and update the VM:

```
$vm.StorageProfile.OSDisk.DiskSizeGB = 1023  
Update-AzVM -ResourceGroupName $rgName -VM $vm
```

#### WARNING

The new size should be greater than the existing disk size. The maximum allowed is 2,048 GB for OS disks. (It's possible to expand the VHD blob beyond that size, but the OS will only be able to work with the first 2,048 GB of space.)

- Update the size of any data disks you want to resize. To expand the first data disk attached to the VM, use a numeric index to obtain a reference to first attached data disk:

```
$vm.StorageProfile.DataDisks[0].DiskSizeGB = 1023
```

Similarly, you can reference other data disks attached to the VM, either by using an index or the **Name** property of the disk:

```
($vm.StorageProfile.DataDisks | Where ($_.Name -eq 'my-second-data-disk')).DiskSizeGB = 1023
```

- Updating the VM might take a few seconds. When the command finishes executing, restart the VM:

```
Start-AzVM -ResourceGroupName $rgName -Name $vmName
```

## Next steps

You can also attach disks using the [Azure portal](#).

# Using disks in Azure Resource Manager Templates

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This document walks through the differences between managed and unmanaged disks when using Azure Resource Manager templates to provision virtual machines. The examples help you to update existing templates that are using unmanaged Disks to managed disks. For reference, we are using the [vm-simple-windows](#) template as a guide. You can see the template using both [managed Disks](#) and a prior version using [unmanaged disks](#) if you'd like to directly compare them.

## Unmanaged Disks template formatting

To begin, let's take a look at how unmanaged disks are deployed. When creating unmanaged disks, you need a storage account to hold the VHD files. You can create a new storage account or use one that already exists. This article shows you how to create a new storage account. Create a storage account resource in the resources block as shown below.

```
{
  "type": "Microsoft.Storage/storageAccounts",
  "apiVersion": "2018-07-01",
  "name": "[variables('storageAccountName')]",
  "location": "[resourceGroup().location]",
  "sku": {
    "name": "Standard_LRS"
  },
  "kind": "Storage",
  "properties": {}
}
```

Within the virtual machine object, add a dependency on the storage account to ensure that it's created before the virtual machine. Within the `storageProfile` section, specify the full URI of the VHD location, which references the storage account and is needed for the OS disk and any data disks.

```
{
  "type": "Microsoft.Compute/virtualMachines",
  "apiVersion": "2018-10-01",
  "name": "[variables('vmName')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[resourceId('Microsoft.Storage/storageAccounts/', variables('storageAccountName'))]",
    "[resourceId('Microsoft.Network/networkInterfaces/', variables('nicName'))]"
  ],
  "properties": {
    "hardwareProfile": {...},
    "osProfile": {...},
    "storageProfile": {
      "imageReference": {
        "publisher": "MicrosoftWindowsServer",
        "offer": "WindowsServer",
        "sku": "[parameters('windowsOSVersion')]",
        "version": "latest"
      },
      "osDisk": {
        "name": "osdisk",
        "vhd": {
          "uri": "[concat(reference(resourceId('Microsoft.Storage/storageAccounts/',
variables('storageAccountName'))).primaryEndpoints.blob, 'vhds/osdisk.vhd')]"
        },
        "caching": "ReadWrite",
        "createOption": "FromImage"
      },
      "dataDisks": [
        {
          "name": "datadisk1",
          "diskSizeGB": 1023,
          "lun": 0,
          "vhd": {
            "uri": "[concat(reference(resourceId('Microsoft.Storage/storageAccounts/',
variables('storageAccountName'))).primaryEndpoints.blob, 'vhds/datadisk1.vhd')]"
          },
          "createOption": "Empty"
        }
      ]
    },
    "networkProfile": {...},
    "diagnosticsProfile": {...}
  }
}
```

## Managed disks template formatting

With Azure Managed Disks, the disk becomes a top-level resource and no longer requires a storage account to be created by the user. Managed disks were first exposed in the `2016-04-30-preview` API version, they are available in all subsequent API versions and are now the default disk type. The following sections walk through the default settings and detail how to further customize your disks.

### NOTE

It is recommended to use an API version later than `2016-04-30-preview` as there were breaking changes between `2016-04-30-preview` and `2017-03-30`.

### Default managed disk settings

To create a VM with managed disks, you no longer need to create the storage account resource. Referencing the template example below, there are some differences from the previous unmanaged disk examples to note:

- The `apiVersion` is a version that supports managed disks.
- `osDisk` and `dataDisks` no longer refer to a specific URI for the VHD.
- When deploying without specifying additional properties, the disk will use a storage type based on the size of the VM. For example, if you are using a VM size that supports premium storage (sizes with "s" in their name such as Standard\_D2s\_v3) then premium disks will be configured by default. You can change this by using the sku setting of the disk to specify a storage type.
- If no name for the disk is specified, it takes the format of `<VMName>_OsDisk_1_<randomstring>` for the OS disk and `<VMName>_disk<#>_<randomstring>` for each data disk.
  - If a VM is being created from a custom image then the default settings for storage account type and disk name are retrieved from the disk properties defined in the custom image resource. These can be overridden by specifying values for these in the template.
- By default, Azure disk encryption is disabled.
- By default, disk caching is Read/Write for the OS disk and None for data disks.
- In the example below there is still a storage account dependency, though this is only for storage of diagnostics and is not needed for disk storage.

```
{
  "type": "Microsoft.Compute/virtualMachines",
  "apiVersion": "2018-10-01",
  "name": "[variables('vmName')]",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[resourceId('Microsoft.Storage/storageAccounts/', variables('storageAccountName'))]",
    "[resourceId('Microsoft.Network/networkInterfaces/', variables('nicName'))]"
  ],
  "properties": {
    "hardwareProfile": {...},
    "osProfile": {...},
    "storageProfile": {
      "imageReference": {
        "publisher": "MicrosoftWindowsServer",
        "offer": "WindowsServer",
        "sku": "[parameters('windowsOSVersion')]",
        "version": "latest"
      },
      "osDisk": {
        "createOption": "FromImage"
      },
      "dataDisks": [
        {
          "diskSizeGB": 1023,
          "lun": 0,
          "createOption": "Empty"
        }
      ]
    },
    "networkProfile": {...},
    "diagnosticsProfile": {...}
  }
}
```

## Using a top-level managed disk resource

As an alternative to specifying the disk configuration in the virtual machine object, you can create a top-level disk resource and attach it as part of the virtual machine creation. For example, you can create a disk resource as follows to use as a data disk.

```
{
    "type": "Microsoft.Compute/disks",
    "apiVersion": "2018-06-01",
    "name": "[concat(variables('vmName'), '-datadisk1')]",
    "location": "[resourceGroup().location]",
    "sku": {
        "name": "Standard_LRS"
    },
    "properties": {
        "creationData": {
            "createOption": "Empty"
        },
        "diskSizeGB": 1023
    }
}
```

Within the VM object, reference the disk object to be attached. Specifying the resource ID of the managed disk created in the `managedDisk` property allows the attachment of the disk as the VM is created. The `apiVersion` for the VM resource is set to `2017-03-30`. A dependency on the disk resource is added to ensure it's successfully created before VM creation.

```
{
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2018-10-01",
    "name": "[variables('vmName')]",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[resourceId('Microsoft.Storage/storageAccounts/', variables('storageAccountName'))]",
        "[resourceId('Microsoft.Network/networkInterfaces/', variables('nicName'))]",
        "[resourceId('Microsoft.Compute/disks/', concat(variables('vmName'), '-datadisk1'))]"
    ],
    "properties": {
        "hardwareProfile": {...},
        "osProfile": {...},
        "storageProfile": {
            "imageReference": {
                "publisher": "MicrosoftWindowsServer",
                "offer": "WindowsServer",
                "sku": "[parameters('windowsOSVersion')]",
                "version": "latest"
            },
            "osDisk": {
                "createOption": "FromImage"
            },
            "dataDisks": [
                {
                    "lun": 0,
                    "name": "[concat(variables('vmName'), '-datadisk1')]",
                    "createOption": "attach",
                    "managedDisk": {
                        "id": "[resourceId('Microsoft.Compute/disks/', concat(variables('vmName'), '-datadisk1'))]"
                    }
                }
            ]
        },
        "networkProfile": {...},
        "diagnosticsProfile": {...}
    }
}
```

## Create managed availability sets with VMs using managed disks

To create managed availability sets with VMs using managed disks, add the `sku` object to the availability set

resource and set the `name` property to `Aligned`. This property ensures that the disks for each VM are sufficiently isolated from each other to avoid single points of failure. Also note that the `apiVersion` for the availability set resource is set to `2018-10-01`.

```
{  
    "type": "Microsoft.Compute/availabilitySets",  
    "apiVersion": "2018-10-01",  
    "location": "[resourceGroup().location]",  
    "name": "[variables('avSetName')]",  
    "properties": {  
        "PlatformUpdateDomainCount": 3,  
        "PlatformFaultDomainCount": 2  
    },  
    "sku": {  
        "name": "Aligned"  
    }  
}
```

## Standard SSD disks

Below are the parameters needed in the Resource Manager template to create Standard SSD Disks:

- `apiVersion` for Microsoft.Compute must be set as `2018-04-01` (or later)
- Specify `managedDisk.storageAccountType` as `StandardSSD_LRS`

The following example shows the `properties.storageProfile.osDisk` section for a VM that uses Standard SSD Disks:

```
"osDisk": {  
    "osType": "Windows",  
    "name": "myOsDisk",  
    "caching": "ReadWrite",  
    "createOption": "FromImage",  
    "managedDisk": {  
        "storageAccountType": "StandardSSD_LRS"  
    }  
}
```

For a complete template example of how to create a Standard SSD disk with a template, see [Create a VM from a Windows Image with Standard SSD Data Disks](#).

## Additional scenarios and customizations

To find full information on the REST API specifications, please review the [create a managed disk REST API documentation](#). You will find additional scenarios, as well as default and acceptable values that can be submitted to the API through template deployments.

## Next steps

- For full templates that use managed disks visit the following Azure Quickstart Repo links.
  - [Windows VM with managed disk](#)
  - [Linux VM with managed disk](#)
- Visit the [Azure Managed Disks Overview](#) document to learn more about managed disks.
- Review the template reference documentation for virtual machine resources by visiting the [Microsoft.Compute/virtualMachines template reference](#) document.
- Review the template reference documentation for disk resources by visiting the [Microsoft.Compute/disks template reference](#) document.
- For information on how to use managed disks in Azure virtual machine scale sets, visit the [Use data disks](#)

[with scale sets](#) document.

# Use Azure Storage Explorer to manage Azure managed disks

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure Storage Explorer contains a rich set of features that allows you to:

- Upload, download, and copy managed disks.
- Create snapshots from operating system or data disk virtual hard disk.
- Migrate data from on-premises to Azure.
- Migrate data across Azure regions.

## Prerequisites

To complete this article, you'll need:

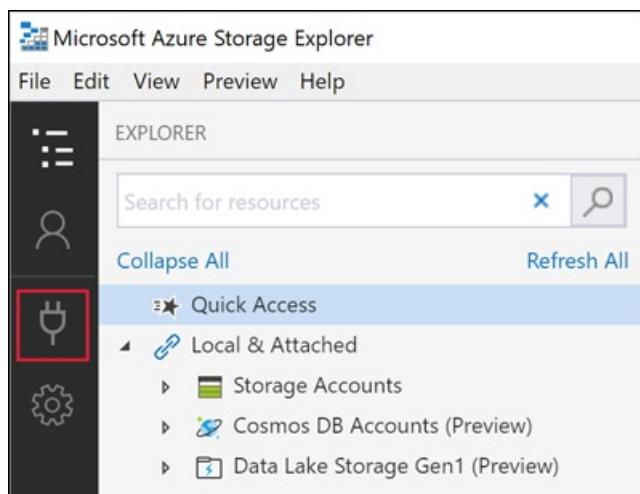
- An Azure subscription.
- At least one Azure managed disk.
- The latest version of [Azure Storage Explorer](#).

If you don't have an Azure subscription, create a [free account](#) before you begin.

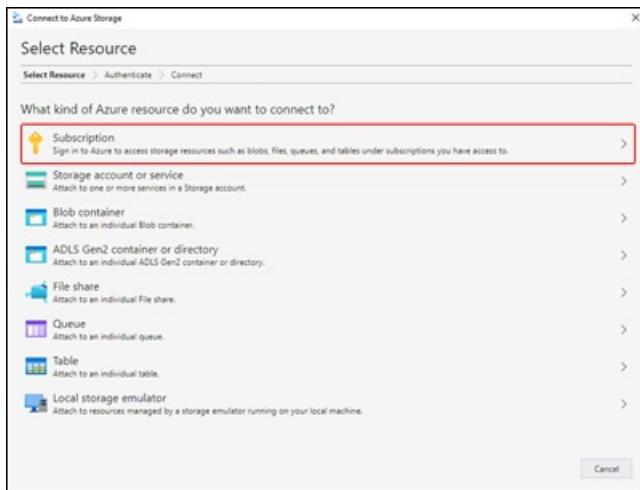
## Connect to an Azure subscription

If your Storage Explorer isn't connected to Azure, you can't use it to manage resources. Follow the steps in this section to connect Storage Explorer to your Azure account. Afterward, you can use it to manage your disks.

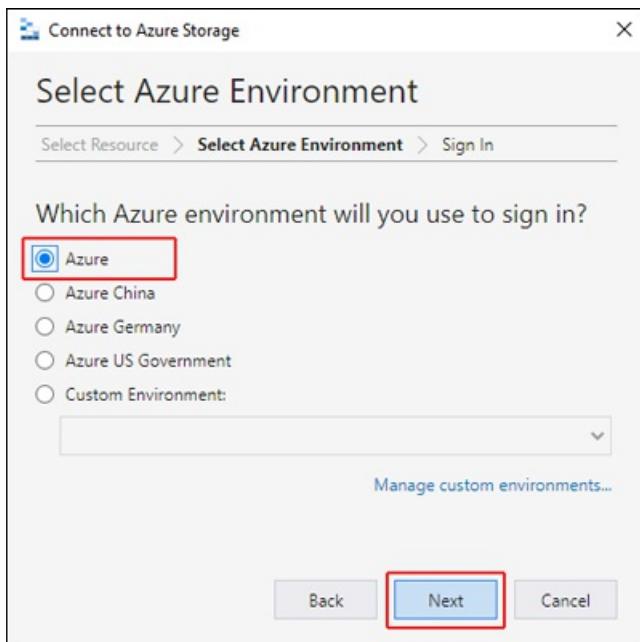
1. Open Azure Storage Explorer and select the **Connect** icon in the toolbar.



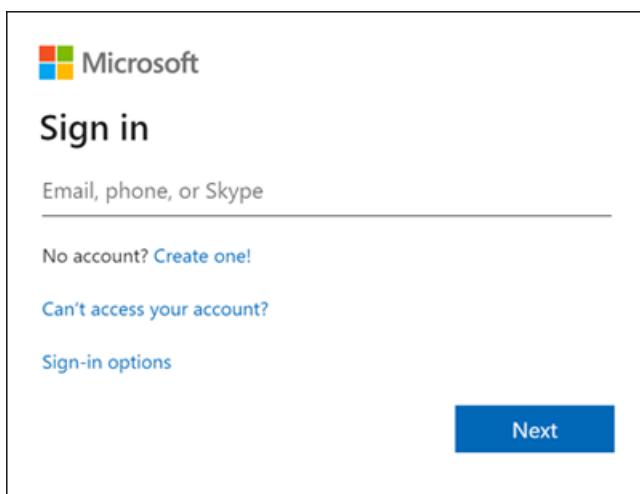
2. In the **Connect to Azure Storage** dialog box, select **Subscription**.



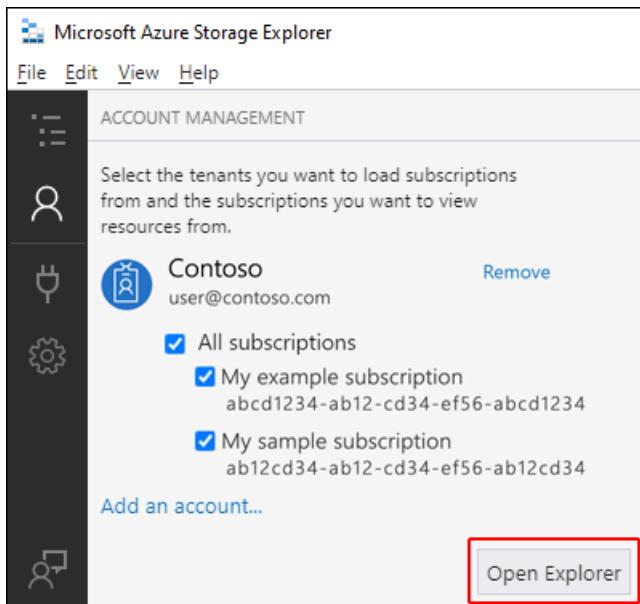
3. Select the appropriate environment and select **Next**. You can also select **Manage custom environments** to configure and add a custom environment.



4. In the **Sign in** dialog box, enter your Azure credentials.



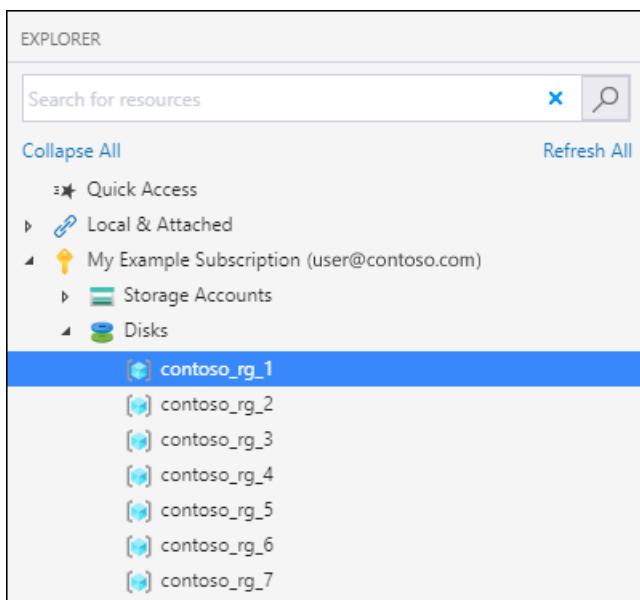
5. Select your subscription from the list and then select **Open Explorer**.



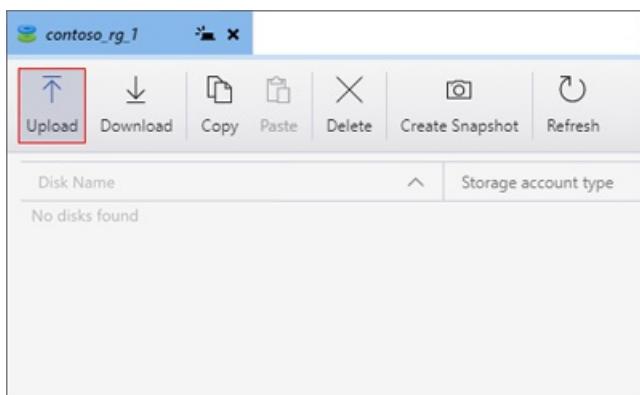
## Upload an on-premises VHD

You can upload an on-premises virtual hard disk (VHD) file to Azure and use it to create an image. Follow the steps in this section to upload your source file.

1. In the **Explorer** pane, expand **Disks** and select the resource group to which you'll upload your disk.

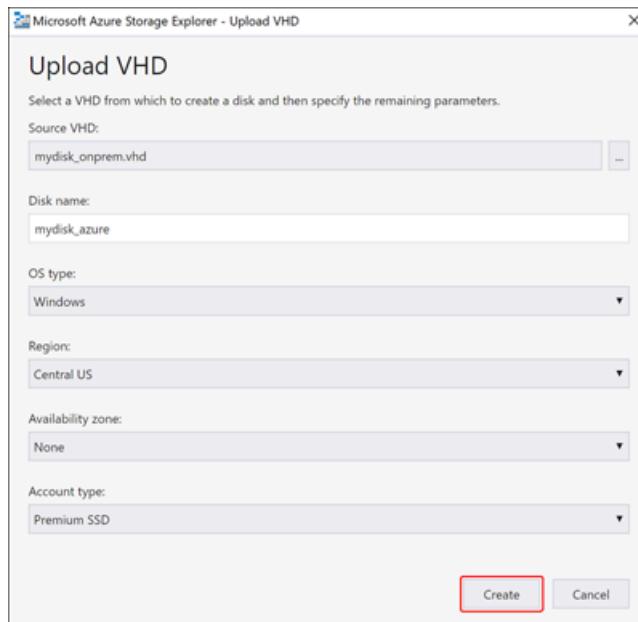


2. In the resource group details pane, select **Upload**.

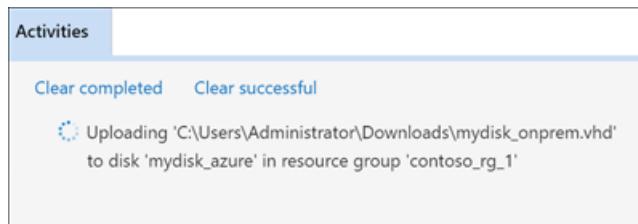


3. In the **Upload VHD** dialog box, specify your VHD source file, the name of the disk, the operating system type, the region to which you want to upload the disk, and the account type. If the region supports

availability zones, you can select a zone of your choice. Select **Create** to begin uploading your disk.



4. The status of the upload will now display in Activities.

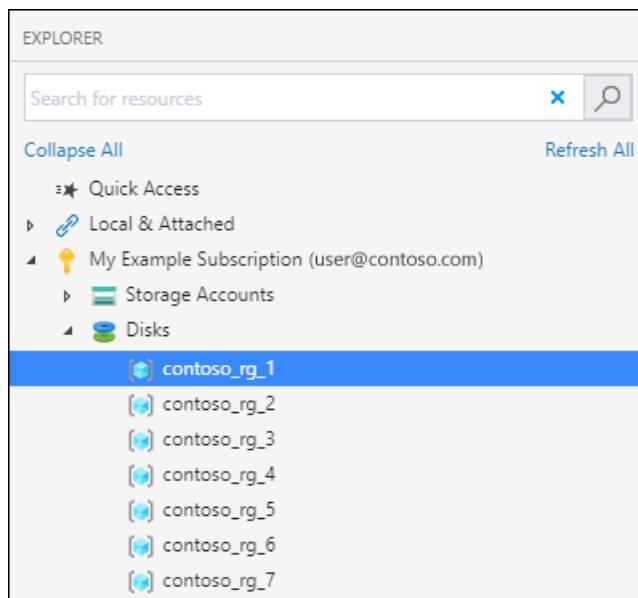


If the upload has finished and you don't see the disk in the Activities pane, select **Refresh**.

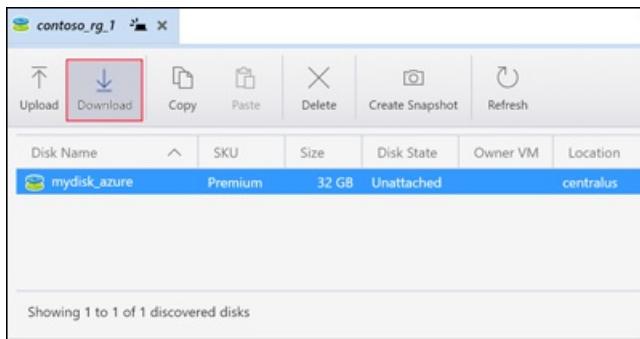
## Download a managed disk

Follow the steps in this section to download a managed disk to an on-premises VHD. A disk's state must be **Unattached** before it can be downloaded.

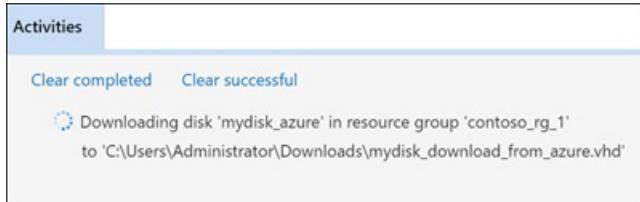
1. In the **Explorer** pane, expand **Disks** and select the resource group from which you'll download your disk.



2. In the resource group details pane, select the disk you want to download.
3. Select **Download** and then choose where you would like to save the disk.



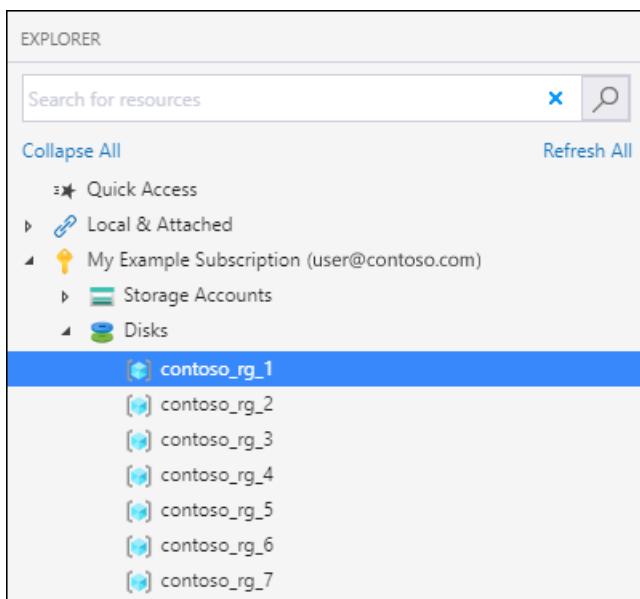
4. Select **Save** to begin the download. The download status will display in **Activities**.



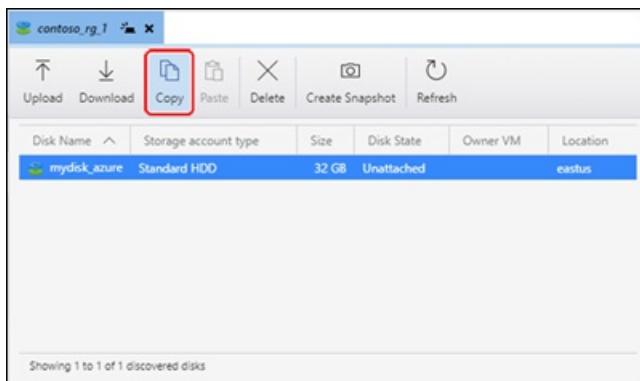
## Copy a managed disk

With Storage Explorer, you can copy a managed disk within or across regions. To copy a disk:

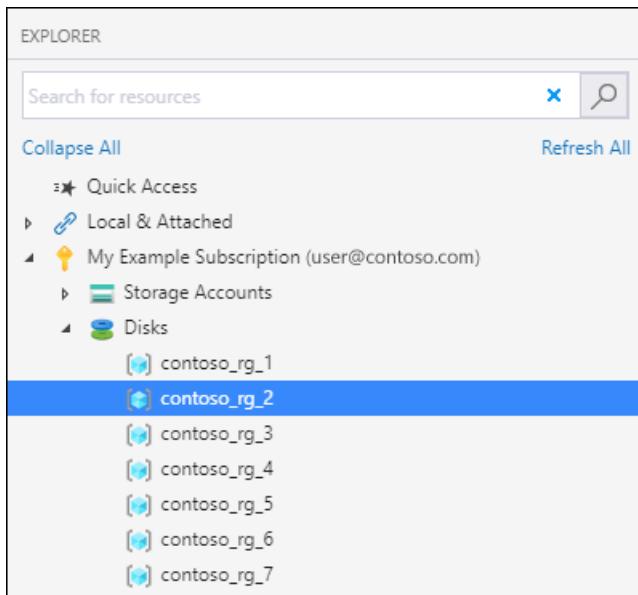
1. In the **Explorer** pane, expand the **Disks** dropdown and select the resource group that contains the disk you want to copy.



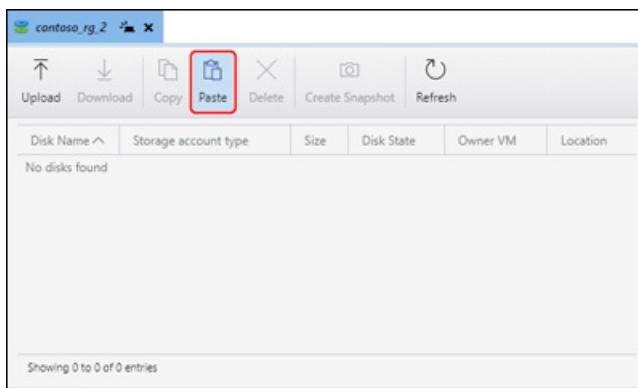
2. In the resource group details pane, select the disk you'd like to copy and select **Copy**.



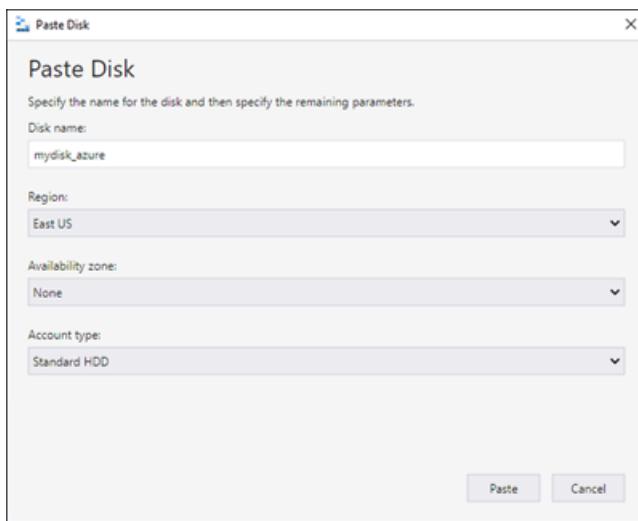
3. In the **Explorer** pane, expand **Disks** and select the resource group in which you'd like to paste the disk.



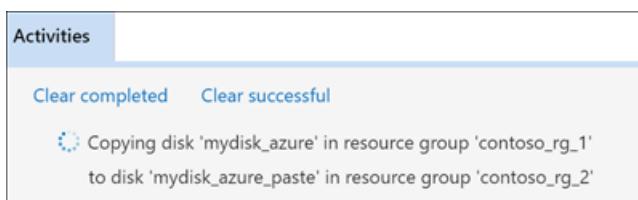
4. Select **Paste** in the resource group details pane.



5. In the **Paste Disk** dialog box, fill in the values. You can also specify an availability zone in supported regions.

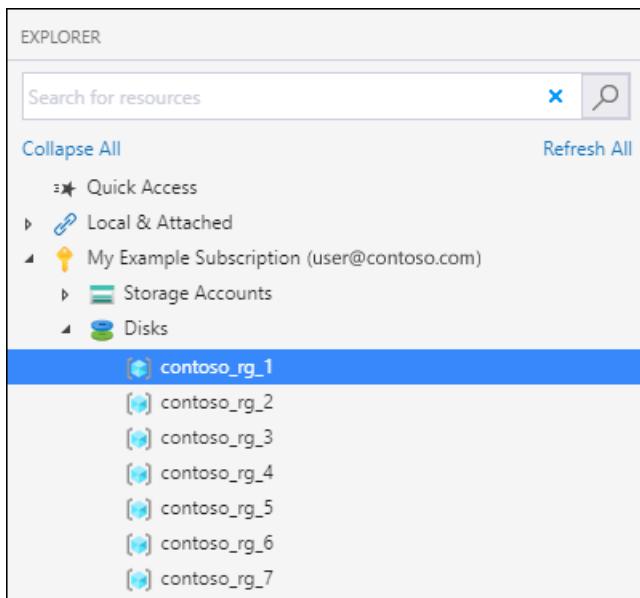


6. Select **Paste** to begin the disk copy. The status is displayed in **Activities**.

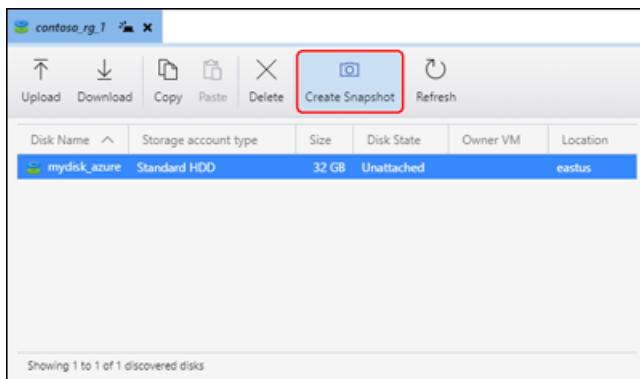


## Create a snapshot

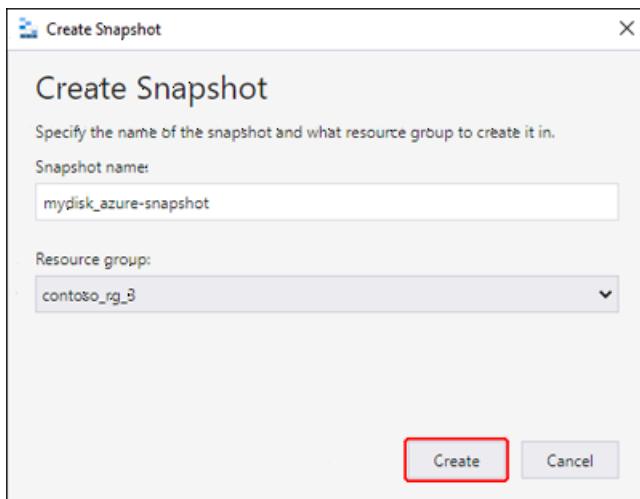
1. In the **Explorer** pane, expand **Disks** and select the resource group that contains the disk you want to snapshot.



2. In the resource group details pane, select the disk you'd like to snapshot and select **Create Snapshot**.



3. In **Create Snapshot**, specify the name of the snapshot and the resource group in which you'll create it. Select **Create**.



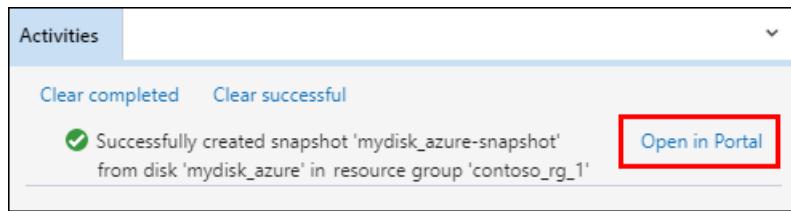
4. After the snapshot has been created, you can select **Open in Portal** in **Activities** to view the snapshot in the Azure portal.

Activities

Clear completed   Clear successful

Successfully created snapshot 'mydisk\_azure-snapshot'  
from disk 'mydisk\_azure' in resource group 'contoso\_rg\_1'

[Open in Portal](#)



The screenshot shows the 'Activities' blade in the Azure portal. It displays a single activity: 'Successfully created snapshot 'mydisk\_azure-snapshot' from disk 'mydisk\_azure' in resource group 'contoso\_rg\_1''. Below the message is a blue link labeled 'Open in Portal'. A red box highlights this link.

## Next steps

- [Create a virtual machine from a VHD by using the Azure portal](#)
- [Attach a managed data disk to a Windows virtual machine by using the Azure portal](#)

# Change the OS disk used by an Azure VM using the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

If you have an existing VM, but you want to swap the disk for a backup disk or another OS disk, you can use the Azure CLI to swap the OS disks. You don't have to delete and recreate the VM. You can even use a managed disk in another resource group, as long as it isn't already in use.

The VM does not need to be stopped\deallocated. The resource ID of the managed disk can be replaced with the resource ID of a different managed disk.

Make sure that the VM size and storage type are compatible with the disk you want to attach. For example, if the disk you want to use is in Premium Storage, then the VM needs to be capable of Premium Storage (like a DS-series size).

This article requires Azure CLI version 2.0.25 or greater. Run `az --version` to find the version. If you need to install or upgrade, see [Install Azure CLI](#).

Use [az disk list](#) to get a list of the disks in your resource group.

```
az disk list \
-g myResourceGroupDisk \
--query '[*].{diskId:id}' \
--output table
```

(Optional) Use [az vm stop](#) to stop\deallocate the VM before swapping the disks.

```
az vm stop \
-n myVM \
-g myResourceGroup
```

Use [az vm update](#) with the full resource ID of the new disk for the `--osdisk` parameter

```
az vm update \
-g myResourceGroup \
-n myVM \
--os-disk /subscriptions/<subscription ID>/resourceGroups/<resource group>/providers/Microsoft.Compute/disks/myDisk
```

Restart the VM using [az vm start](#).

```
az vm start \
-n myVM \
-g myResourceGroup
```

## Next steps

To create a copy of a disk, see [Snapshot a disk](#).

# Change the OS disk used by an Azure VM using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

If you have an existing VM, but you want to swap the disk for a backup disk or another OS disk, you can use Azure PowerShell to swap the OS disks. You don't have to delete and recreate the VM. You can even use a managed disk in another resource group, as long as it isn't already in use.

The VM does not need to be stopped\deallocated. The resource ID of the managed disk can be replaced with the resource ID of a different managed disk.

Make sure that the VM size and storage type are compatible with the disk you want to attach. For example, if the disk you want to use is in Premium Storage, then the VM needs to be capable of Premium Storage (like a DS-series size). Both disks must also be the same size. And ensure that you're not mixing an un-encrypted VM with an encrypted OS disk, this is not supported. If the VM doesn't use Azure Disk Encryption, then the OS disk being swapped in shouldn't be using Azure Disk Encryption. If disks are using Disk Encryption Sets, both disks should belong to same Disk Encryption set.

Get a list of disks in a resource group using [Get-AzDisk](#)

```
Get-AzDisk -ResourceGroupName myResourceGroup | Format-Table -Property Name
```

When you have the name of the disk that you would like to use, set that as the OS disk for the VM. This example stop\deallocates the VM named *myVM* and assigns the disk named *newDisk* as the new OS disk.

```
# Get the VM
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM

# (Optional) Stop/ deallocate the VM
Stop-AzVM -ResourceGroupName myResourceGroup -Name $vm.Name -Force

# Get the new disk that you want to swap in
$disk = Get-AzDisk -ResourceGroupName myResourceGroup -Name newDisk

# Set the VM configuration to point to the new disk
Set-AzVMOSDisk -VM $vm -ManagedDiskId $disk.Id -Name $disk.Name

# Update the VM with the new OS disk
Update-AzVM -ResourceGroupName myResourceGroup -VM $vm

# Start the VM
Start-AzVM -Name $vm.Name -ResourceGroupName myResourceGroup
```

## Next steps

To create a copy of a disk, see [Snapshot a disk](#).

# How to map Azure Disks to Linux VM guest disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You may need to determine the Azure Disks that back a VM's guest disks. In some scenarios, you can compare the disk or volume size to the size of the attached Azure Disks. In scenarios where there are multiple Azure Disks of the same size attached to the VM you need to use the Logical Unit Number (LUN) of the data disks.

## What is a LUN?

A Logical Unit Number (LUN) is a number that is used to identify a specific storage device. Each storage device is assigned a unique numeric identifier, starting at zero. The full path to a device is represented by the bus number, target ID number, and Logical Unit Number (LUN).

For example: ***Bus Number 0, Target ID 0, LUN 3***

For our exercise, you only need to use the LUN.

## Finding the LUN

Below we have listed two methods for finding the LUN of a disk in Linux.

### **lsscsi**

1. Connect to the VM
2. `sudo lsscsi`

The first column listed will contain the LUN, the format is [Host:Channel:Target:LUN].

### **Listing block devices**

1. Connect to the VM
2. `sudo ls -l /sys/block/*/device`

The last column listed will contain the LUN, the format is [Host:Channel:Target:LUN]

## Finding the LUN for the Azure Disks

You can locate the LUN for an Azure Disk using the Azure portal, Azure CLI.

### **Finding an Azure Disk's LUN in the Azure portal**

1. In the Azure portal, select "Virtual Machines" to display a list of your Virtual Machines
2. Select the Virtual Machine
3. Select "Disks"
4. Select a data disk from the list of attached disks.
5. The LUN of the disk will be displayed in the disk detail pane. The LUN displayed here correlate to the LUNs that you looked up in the Guest using lsscsi, or listing the block devices.

### **Finding an Azure Disk's LUN using Azure CLI**

```
az vm show -g myResourceGroup -n myVM --query "storageProfile.dataDisks"
```

# How to map Azure Disks to Windows VM guest disks

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

You may need to determine the Azure Disks that back a VM's guest disks. In some scenarios, you can compare the disk or volume size to the size of the attached Azure Disks. In scenarios where there are multiple Azure Disks of the same size attached to the VM you need to use the Logical Unit Number (LUN) of the data disks.

## What is a LUN?

A Logical Unit Number (LUN) is a number that is used to identify a specific storage device. Each storage device is assigned a unique numeric identifier, starting at zero. The full path to a device is represented by the bus number, target ID number, and Logical Unit Number (LUN).

For example: ***Bus Number 0, Target ID 0, LUN 3***

For our exercise, you only need to use the LUN.

## Finding the LUN

There are two methods to finding the LUN, which one you choose will depend on if you are using [Storage Spaces](#) or not.

### Disk Management

If you are not using Storage Pools, you can use [Disk Management](#) to find the LUN.

1. Connect to the VM and open Disk Management a. Right-click on the Start button and choose "Disk Management" a. You can also type `diskmgmt.msc` into the Start Search box
2. In the lower pane, right-click any of the Disks and choose "Properties"
3. The LUN will be listed in the "Location" property on the "General" tab

### Storage Pools

1. Connect to the VM and open Server Manager
2. Select "File and Storage Services", "Volumes", "Storage Pools"
3. In the bottom-right corner of Server Manager, there will be a "Physical Disks" section. The disks that make up the Storage Pool are listed here as well as the LUN for each disk.

## Finding the LUN for the Azure Disks

You can locate the LUN for an Azure Disk using the Azure portal, Azure CLI, or Azure PowerShell

### Finding an Azure Disk's LUN in the Azure portal

1. In the Azure portal, select "Virtual Machines" to display a list of your Virtual Machines
2. Select the Virtual Machine
3. Select "Disks"
4. Select a data disk from the list of attached disks.
5. The LUN of the disk will be displayed in the disk detail pane. The LUN displayed here correlates to the LUNs that were looked up in the Guest using Device Manager or Server Manager.

## Finding an Azure Disk's LUN using Azure CLI or Azure PowerShell

- [Azure CLI](#)
- [Azure PowerShell](#)

```
az vm show -g myResourceGroup -n myVM --query "storageProfile.dataDisks"
```

# Use SCP to move files to and from a VM

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article shows how to move files from your workstation up to an Azure VM, or from an Azure VM down to your workstation, using Secure Copy (SCP). Moving files between your workstation and a VM, quickly and securely, is critical for managing your Azure infrastructure.

For this article, you need a VM deployed in Azure with SSH enabled. You also need an SCP client for your local computer. It is built on top of SSH and included in the default shell of most computers.

## Quick commands

Copy a file up to the VM

```
scp file azureuser@azurehost:directory/targetfile
```

Copy a file down from the VM

```
scp azureuser@azurehost:directory/file targetfile
```

## Detailed walkthrough

As examples, we move an Azure configuration file up to a VM and pull down a log file directory, both using SCP.

## SSH key pair authentication

SCP uses SSH for the transport layer. SSH handles the authentication on the destination host, and it moves the file in an encrypted tunnel provided by default with SSH. For SSH authentication, usernames and passwords can be used. However, SSH public and private key authentication are recommended as a security best practice. Once SSH has authenticated the connection, SCP then begins copying the file. Using a properly configured `~/.ssh/config` and SSH public and private keys, the SCP connection can be established by just using a server name (or IP address). If you only have one SSH key, SCP looks for it in the `~/.ssh/` directory, and uses it by default to log in to the VM.

For more information on configuring your `~/.ssh/config` and SSH public and private keys, see [Create SSH keys](#).

## SCP a file to a VM

For the first example, we copy an Azure configuration file up to a VM that is used to deploy automation. Because this file contains Azure API credentials, which include secrets, security is important. The encrypted tunnel provided by SSH protects the contents of the file.

The following command copies the local `.azure/config` file to an Azure VM with FQDN `myserver.eastus.cloudapp.azure.com`. If you don't have an [FQDN set](#), you can also use the IP address of the VM. The admin user name on the Azure VM is `azureuser`. The file is targeted to the `/home/azureuser/` directory. Substitute your own values in this command.

```
scp ~/azure/config azureuser@myserver.eastus.cloudapp.com:/home/azureuser/config
```

## SCP a directory from a VM

For this example, we copy a directory of log files from the VM down to your workstation. A log file may or may not contain sensitive or secret data. However, using SCP ensures the contents of the log files are encrypted. Using SCP to transfer the files is the easiest way to get the log directory and files down to your workstation while also being secure.

The following command copies files in the `/home/azureuser/logs/` directory on the Azure VM to the local `/tmp` directory:

```
scp -r azureuser@myserver.eastus.cloudapp.com:/home/azureuser/logs/. /tmp/
```

The `-r` flag instructs SCP to recursively copy the files and directories from the point of the directory listed in the command. Also notice that the command-line syntax is similar to a `cp` copy command.

## Next steps

- [Manage users, SSH, and check or repair disks on Azure Linux VMs using the VMAccess Extension](#)

# Find and delete unattached Azure managed and unmanaged disks using the Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

When you delete a virtual machine (VM) in Azure, by default, any disks that are attached to the VM aren't deleted. This feature helps to prevent data loss due to the unintentional deletion of VMs. After a VM is deleted, you will continue to pay for unattached disks. This article shows you how to find and delete any unattached disks and reduce unnecessary costs.

## Managed disks: Find and delete unattached disks

The following script looks for unattached [managed disks](#) by examining the value of the **ManagedBy** property. When a managed disk is attached to a VM, the **ManagedBy** property contains the resource ID of the VM. When a managed disk is unattached, the **ManagedBy** property is null. The script examines all the managed disks in an Azure subscription. When the script locates a managed disk with the **ManagedBy** property set to null, the script determines that the disk is unattached.

### IMPORTANT

First, run the script by setting the **deleteUnattachedDisks** variable to 0. This action lets you find and view all the unattached managed disks.

After you review all the unattached disks, run the script again and set the **deleteUnattachedDisks** variable to 1. This action lets you delete all the unattached managed disks.

```
# Set deleteUnattachedDisks=1 if you want to delete unattached Managed Disks
# Set deleteUnattachedDisks=0 if you want to see the Id of the unattached Managed Disks
deleteUnattachedDisks=0
unattachedDiskIds=$(az disk list --query '[?managedBy==`null`].[id]' -o tsv)
for id in ${unattachedDiskIds[@]}
do
    if (( $deleteUnattachedDisks == 1 ))
    then
        echo "Deleting unattached Managed Disk with Id: "$id
        az disk delete --ids $id --yes
        echo "Deleted unattached Managed Disk with Id: "$id
    else
        echo $id
    fi
done
```

## Unmanaged disks: Find and delete unattached disks

Unmanaged disks are VHD files that are stored as [page blobs](#) in [Azure storage accounts](#). The following script looks for unattached unmanaged disks (page blobs) by examining the value of the **LeaseStatus** property. When an unmanaged disk is attached to a VM, the **LeaseStatus** property is set to **Locked**. When an unmanaged disk is unattached, the **LeaseStatus** property is set to **Unlocked**. The script examines all the unmanaged disks in all the Azure storage accounts in an Azure subscription. When the script locates an unmanaged disk with a

**LeaseStatus** property set to **Unlocked**, the script determines that the disk is unattached.

#### IMPORTANT

First, run the script by setting the **deleteUnattachedVHDs** variable to 0. This action lets you find and view all the unattached unmanaged VHDs.

After you review all the unattached disks, run the script again and set the **deleteUnattachedVHDs** variable to 1. This action lets you delete all the unattached unmanaged VHDs.

```
# Set deleteUnattachedVHDs=1 if you want to delete unattached VHDs
# Set deleteUnattachedVHDs=0 if you want to see the details of the unattached VHDs
deleteUnattachedVHDs=0
storageAccountIds=$(az storage account list --query [].[id] -o tsv)
for id in ${storageAccountIds[@]}
do
    connectionString=$(az storage account show-connection-string --ids $id --query connectionString -o tsv)
    containers=$(az storage container list --connection-string $connectionString --query [].[name] -o tsv)

    for container in ${containers[@]}
    do

        blobs=$(az storage blob list --show-next-marker -c $container --connection-string $connectionString
--query "[?properties.blobType=='PageBlob' && ends_with(name, '.vhd')].[name]" -o tsv)

        for blob in ${blobs[@]}
        do
            leaseStatus=$(az storage blob show -n $blob -c $container --connection-string $connectionString
--query "properties.lease.status" -o tsv)

            if [ "$leaseStatus" == "unlocked" ]
            then

                if (( $deleteUnattachedVHDs == 1 ))
                then

                    echo "Deleting VHD: \"$blob\" in container: \"$container\" in storage account: \"$id"

                    az storage blob delete --delete-snapshots include -n $blob -c $container --connection-
string $connectionString

                    echo "Deleted VHD: \"$blob\" in container: \"$container\" in storage account: \"$id"
                else
                    echo "StorageAccountId: \"$id\" container: \"$container\" VHD: \"$blob"
                fi

            fi
        done
    done
done
```

## Next steps

For more information, see [Delete a storage account](#).

# Find and delete unattached Azure managed and unmanaged disks

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

When you delete a virtual machine (VM) in Azure, by default, any disks that are attached to the VM aren't deleted. This feature helps to prevent data loss due to the unintentional deletion of VMs. After a VM is deleted, you will continue to pay for unattached disks. This article shows you how to find and delete any unattached disks and reduce unnecessary costs.

## Managed disks: Find and delete unattached disks

The following script looks for unattached [managed disks](#) by examining the value of the **ManagedBy** property. When a managed disk is attached to a VM, the **ManagedBy** property contains the resource ID of the VM. When a managed disk is unattached, the **ManagedBy** property is null. The script examines all the managed disks in an Azure subscription. When the script locates a managed disk with the **ManagedBy** property set to null, the script determines that the disk is unattached.

### IMPORTANT

First, run the script by setting the **deleteUnattachedDisks** variable to 0. This action lets you find and view all the unattached managed disks.

After you review all the unattached disks, run the script again and set the **deleteUnattachedDisks** variable to 1. This action lets you delete all the unattached managed disks.

```
# Set deleteUnattachedDisks=1 if you want to delete unattached Managed Disks
# Set deleteUnattachedDisks=0 if you want to see the Id of the unattached Managed Disks
$deleteUnattachedDisks=0
$managedDisks = Get-AzDisk
foreach ($md in $managedDisks) {
    # ManagedBy property stores the Id of the VM to which Managed Disk is attached to
    # If ManagedBy property is $null then it means that the Managed Disk is not attached to a VM
    if($md.ManagedBy -eq $null){
        if($deleteUnattachedDisks -eq 1){
            Write-Host "Deleting unattached Managed Disk with Id: $($md.Id)"
            $md | Remove-AzDisk -Force
            Write-Host "Deleted unattached Managed Disk with Id: $($md.Id) "
        }else{
            $md.Id
        }
    }
}
```

## Unmanaged disks: Find and delete unattached disks

Unmanaged disks are VHD files that are stored as [page blobs](#) in [Azure storage accounts](#). The following script looks for unattached unmanaged disks (page blobs) by examining the value of the **LeaseStatus** property. When an unmanaged disk is attached to a VM, the **LeaseStatus** property is set to **Locked**. When an unmanaged disk is unattached, the **LeaseStatus** property is set to **Unlocked**. The script examines all the unmanaged disks in all the Azure storage accounts in an Azure subscription. When the script locates an unmanaged disk with a

**LeaseStatus** property set to **Unlocked**, the script determines that the disk is unattached.

#### IMPORTANT

First, run the script by setting the **deleteUnattachedVHDs** variable to `$false`. This action lets you find and view all the unattached unmanaged VHDs.

After you review all the unattached disks, run the script again and set the **deleteUnattachedVHDs** variable to `$true`. This action lets you delete all the unattached unmanaged VHDs.

```
# Set deleteUnattachedVHDs=$true if you want to delete unattached VHDs
# Set deleteUnattachedVHDs=$false if you want to see the Uri of the unattached VHDs
$deleteUnattachedVHDs=$false
$storageAccounts = Get-AzStorageAccount
foreach($storageAccount in $storageAccounts){
    $storageKey = (Get-AzStorageAccountKey -ResourceGroupName $storageAccount.ResourceGroupName -Name
    $storageAccount.StorageAccountName)[0].Value
    $context = New-AzStorageContext -StorageAccountName $storageAccount.StorageAccountName -
    StorageAccountKey $storageKey
    $containers = Get-AzStorageContainer -Context $context
    foreach($container in $containers){
        $blobs = Get-AzStorageBlob -Container $container.Name -Context $context
        #Fetch all the Page blobs with extension .vhd as only Page blobs can be attached as disk to Azure
        VMs
        $blobs | Where-Object {$_._BlobType -eq 'PageBlob' -and $_.Name.EndsWith('.vhd')} | ForEach-Object {
            #If a Page blob is not attached as disk then LeaseStatus will be unlocked
            if($_.ICloudBlob.Properties.LeaseStatus -eq 'Unlocked'){
                if($deleteUnattachedVHDs){
                    Write-Host "Deleting unattached VHD with Uri: $($_.ICloudBlob.Uri.AbsoluteUri)"
                    $_ | Remove-AzStorageBlob -Force
                    Write-Host "Deleted unattached VHD with Uri: $($_.ICloudBlob.Uri.AbsoluteUri)"
                }
                else{
                    $_.ICloudBlob.Uri.AbsoluteUri
                }
            }
        }
    }
}
```

## Next steps

For more information, see [Delete a storage account](#) and [Identify Orphaned Disks Using PowerShell](#)

# Find and delete unattached Azure managed and unmanaged disks - Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

When you delete a virtual machine (VM) in Azure, by default, any disks that are attached to the VM aren't deleted. This helps to prevent data loss due to the unintentional deletion of VMs. After a VM is deleted, you will continue to pay for unattached disks. This article shows you how to find and delete any unattached disks using the Azure portal, and reduce unnecessary costs. Deletions are permanent, you will not be able to recover data once you delete a disk.

## Managed disks: Find and delete unattached disks

If you have unattached managed disks and no longer need the data on them, the following process explains how to find them from the Azure portal:

1. Sign in to the [Azure portal](#).
  2. Search for and select **Disk**.
- On the **Disk** blade, you are presented with a list of all your disks.
3. Select the disk you'd like to delete, this brings you to the individual disk's blade.
  4. On the individual disk's blade, confirm the disk state is unattached, then select **Delete**.

The screenshot shows the 'data-detach' blade for a disk named 'linvm'. At the top, there's a search bar labeled 'Search (Ctrl+ /)' and several action buttons: '+ Create VM', '+ Create snapshot', 'Delete' (which is highlighted with a red box), and 'Refresh'. Below the search bar, there's a navigation menu with 'Overview' selected (also highlighted with a red box). To the right of the menu, it says 'Resource group (change)' followed by 'linvm'. Further down, under 'Disk state', it shows 'Unattached' (also highlighted with a red box).

## Unmanaged disks: Find and delete unattached disks

Unmanaged disks are VHD files that are stored as [page blobs](#) in [Azure storage accounts](#).

If you have unmanaged disks that aren't attached to a VM, no longer need the data on them, and would like to delete them, the following process explains how to do so from the Azure portal:

1. Sign in to the [Azure portal](#).
2. Search for and select **Disk (Classic)**.

You are presented with a list of all your unmanaged disks. Any disk that has "-" in the **Attached to** column is an unattached disk.

## Disks (classic) ☁

Microsoft

[+ Add](#) [Edit columns](#) [⟳ Refresh](#) | [Assign tags](#)

**Subscriptions:** All 24 selected – Don't see a subscription? [Open Directory + Subscription settings](#)

All subscriptions

5 items

<input type="checkbox"/>	Name ↑↓	IO type	Size	Attached to
<input type="checkbox"/>	 asrde...	Standard	32 GiB	-

3. Select the unattached disk you'd like to delete, this brings up the individual disk's blade.

4. On that individual disk's blade, you can confirm it is unattached, since **Attached to** will still be -.

### Attached to (i)

-

5. Select **Delete**.

[+ Create VM](#) [Delete](#)

## Next steps

If you'd like an automated way of finding and deleting unattached storage accounts, see our [CLI](#) or [PowerShell](#) articles.

For more information, see [Delete a storage account](#) and [Identify Orphaned Disks Using PowerShell](#)



# Create a managed disk from a VHD file in a storage account in the same subscription with CLI (Linux)

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script creates a managed disk from a VHD file in a storage account in the same subscription. Use this script to import a specialized (not generalized/sysprepped) VHD to managed OS disk to create a virtual machine. Or, use it to import a data VHD to managed data disk.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id  
subscriptionId=<subscriptionId>  
  
#Provide the name of your resource group.  
#Ensure that resource group is already created  
resourceGroupName=myResourceGroupName  
  
#Provide the name of the Managed Disk  
diskName=myDiskName  
  
#Provide the size of the disks in GB. It should be greater than the VHD file size.  
diskSize=128  
  
#Provide the URI of the VHD file that will be used to create Managed Disk.  
# VHD file can be deleted as soon as Managed Disk is created.  
# e.g. https://contosostorageaccount1.blob.core.windows.net/vhds/contosovhd123.vhd  
vhdUri=https://contosostorageaccount1.blob.core.windows.net/vhds/contosoumd78620170425131836.vhd  
  
#Provide the storage type for the Managed Disk. Premium_LRS or Standard_LRS.  
storageType=Premium_LRS  
  
#Provide the Azure location (e.g. westus) where Managed Disk will be located.  
#The location should be same as the location of the storage account where VHD file is stored.  
#Get all the Azure location supported for your subscription using command below:  
#az account list-locations  
location=westus  
  
#Set the context to the subscription Id where Managed Disk will be created  
az account set --subscription $subscriptionId  
  
#Create the Managed disk from the VHD file  
az disk create --resource-group $resourceGroupName --name $diskName --sku $storageType --location $location  
--size-gb $diskSize --source $vhdUri
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses following commands to create a managed disk from a VHD. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az disk create</a>	Creates a managed disk using URI of a VHD in a storage account in the same subscription

## Next steps

[Create a virtual machine by attaching a managed disk as OS disk](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Create a managed disk from a snapshot with CLI (Linux)

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script creates a managed disk from a snapshot. Use it to restore a virtual machine from snapshots of OS and data disks. Create OS and data managed disks from respective snapshots and then create a new virtual machine by attaching managed disks. You can also restore data disks of an existing VM by attaching data disks created from snapshots.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id of the subscription where you want to create Managed Disks  
subscriptionId=<subscriptionId>"  
  
#Provide the name of your resource group  
resourceGroupName=myResourceGroupName  
  
#Provide the name of the snapshot that will be used to create Managed Disks  
snapshotName=mySnapshotName  
  
#Provide the name of the new Managed Disks that will be created  
diskName=myDiskName  
  
#Provide the size of the disks in GB. It should be greater than the VHD file size.  
diskSize=128  
  
#Provide the storage type for Managed Disk. Premium_LRS or Standard_LRS.  
storageType=Premium_LRS  
  
#Set the context to the subscription Id where Managed Disk will be created  
az account set --subscription $subscriptionId  
  
#Get the snapshot Id  
snapshotId=$(az snapshot show --name $snapshotName --resource-group $resourceGroupName --query [id] -o tsv)  
  
#Create a new Managed Disks using the snapshot Id  
#Note that managed disk will be created in the same location as the snapshot  
az disk create --resource-group $resourceGroupName --name $diskName --sku $storageType --size-gb $diskSize -  
-source $snapshotId
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses following commands to create a managed disk from a snapshot. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az snapshot show</a>	Gets all the properties of a snapshot using the name and resource group properties of the snapshot. Id property is used to create managed disk.
<a href="#">az disk create</a>	Creates a managed disk using snapshot Id of a managed snapshot

## Next steps

[Create a virtual machine by attaching a managed disk as OS disk](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Copy managed disks to same or different subscription with CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script copies a managed disk to same or different subscription but in the same region. The copy works only when the subscriptions are part of the same Azure AD tenant.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.



[Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id of the subscription where managed disk exists  
sourceSubscriptionId=<subscriptionId>  
  
#Provide the name of your resource group where managed disk exists  
sourceResourceGroupName=mySourceResourceGroupName  
  
#Provide the name of the managed disk  
managedDiskName=myDiskName  
  
#Set the context to the subscription Id where managed disk exists  
az account set --subscription $sourceSubscriptionId  
  
#Get the managed disk Id  
managedDiskId=$(az disk show --name $managedDiskName --resource-group $sourceResourceGroupName --query [id] -o tsv)  
  
#If managedDiskId is blank then it means that managed disk does not exist.  
echo 'source managed disk Id is: ' $managedDiskId  
  
#Provide the subscription Id of the subscription where managed disk will be copied to  
targetSubscriptionId=6492b1f7-f219-446b-b509-314e17e1efb0  
  
#Name of the resource group where managed disk will be copied to  
targetResourceGroupName=mytargetResourceGroupName  
  
#Set the context to the subscription Id where managed disk will be copied to  
az account set --subscription $targetSubscriptionId  
  
#Copy managed disk to different subscription using managed disk Id  
az disk create --resource-group $targetResourceGroupName --name $managedDiskName --source $managedDiskId
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name mySourceResourceGroupName
```

## Sample reference

This script uses following commands to create a new managed disk in the target subscription using the **Id** of the source managed disk. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az disk show</a>	Gets all the properties of a managed disk using the name and resource group properties of the managed disk. The <b>Id</b> property is used to copy the managed disk to different subscription.

COMMAND	NOTES
<code>az disk create</code>	Copies a managed disk by creating a new managed disk in different subscription using the <code>Id</code> and name the parent managed disk.

## Next steps

[Create a virtual machine from a managed disk](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Export/Copy a snapshot to a storage account in different region with CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script exports a managed snapshot to a storage account in different region. It first generates the SAS URI of the snapshot and then uses it to copy it to a storage account in different region. Use this script to maintain backup of your managed disks in different region for disaster recovery.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id where snapshot is created  
subscriptionId=<subscriptionId>"  
  
#Provide the name of your resource group where snapshot is created  
resourceGroupName=myResourceGroupName  
  
#Provide the snapshot name  
snapshotName=mySnapshotName  
  
#Provide Shared Access Signature (SAS) expiry duration in seconds e.g. 3600.  
#Know more about SAS here: https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-shared-access-signature-part-1  
sasExpiryDuration=3600  
  
#Provide storage account name where you want to copy the snapshot.  
storageAccountName=mystorageaccountname  
  
#Name of the storage container where the downloaded snapshot will be stored  
storageContainerName=mystoragecontainername  
  
#Provide the key of the storage account where you want to copy snapshot.  
storageAccountKey=mystorageaccountkey  
  
#Provide the name of the VHD file to which snapshot will be copied.  
destinationVHDFilename=myvhdfilename  
  
az account set --subscription $subscriptionId  
  
sas=$(az snapshot grant-access --resource-group $resourceGroupName --name $snapshotName --duration-in-seconds $sasExpiryDuration --query [accessSas] -o tsv)  
  
az storage blob copy start --destination-blob $destinationVHDFilename --destination-container $storageContainerName --account-name $storageAccountName --account-key $storageAccountKey --source-uri $sas
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses following commands to generate SAS URI for a managed snapshot and copies the snapshot to a storage account using SAS URI. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az snapshot grant-access</a>	Generates read-only SAS that is used to copy underlying VHD file to a storage account or download it to on-premises

COMMAND	NOTES
<code>az storage blob copy start</code>	Copies a blob asynchronously from one storage account to another

## Next steps

[Create a managed disk from a VHD](#)

[Create a virtual machine from a managed disk](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Export/Copy a managed disk to a storage account using the Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script exports the underlying VHD of a managed disk to a storage account in same or different region. It first generates the SAS URI of the managed disk and then uses it to copy the VHD to a storage account. Use this script to copy managed disks to another region for regional expansion. If you want to publish the VHD file of a managed disk in Azure Marketplace, you can use this script to copy the VHD file to a storage account and then generate a SAS URI of the copied VHD to publish it in the Marketplace.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id where managed disk is created  
subscriptionId=<subscriptionId>"  
  
#Provide the name of your resource group where managed disk is created  
resourceGroupName=myResourceGroupName  
  
#Provide the managed disk name  
diskName=myDiskName  
  
#Provide Shared Access Signature (SAS) expiry duration in seconds e.g. 3600.  
#Know more about SAS here: https://docs.microsoft.com/en-us/azure/storage/storage-dotnet-shared-access-signature-part-1  
sasExpiryDuration=3600  
  
#Provide storage account name where you want to copy the underlying VHD file of the managed disk.  
storageAccountName=mystorageaccountname  
  
#Name of the storage container where the downloaded VHD will be stored  
storageContainerName=mystoragecontainername  
  
#Provide the key of the storage account where you want to copy the VHD  
storageAccountKey=mystorageaccountkey  
  
#Provide the name of the destination VHD file to which the VHD of the managed disk will be copied.  
destinationVHDFilename=myvhdfilename.vhd  
  
az account set --subscription $subscriptionId  
  
$sas=$(az disk grant-access --resource-group $resourceGroupName --name $diskName --duration-in-seconds  
$sasExpiryDuration --query [accessSas] -o tsv)  
  
az storage blob copy start --destination-blob $destinationVHDFilename --destination-container  
$storageContainerName --account-name $storageAccountName --account-key $storageAccountKey --source-uri $sas
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses following commands to generate the SAS URI for a managed disk and copies the underlying VHD to a storage account using the SAS URI. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az disk grant-access</a>	Generates read-only SAS that is used to copy the underlying VHD file to a storage account or download it to on-premises

COMMAND	NOTES
<code>az storage blob copy start</code>	Copies a blob asynchronously from one storage account to another

## Next steps

[Create a managed disk from a VHD](#)

[Create a virtual machine from a managed disk](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Copy snapshot of a managed disk to same or different subscription with CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script copies a snapshot of a managed disk to same or different subscription. Use this script for the following scenarios:

- Migrate a snapshot in Premium storage (Premium\_LRS) to Standard storage (Standard\_LRS or Standard\_ZRS) to reduce your cost.
- Migrate a snapshot from locally redundant storage (Premium\_LRS, Standard\_LRS) to zone redundant storage (Standard\_ZRS) to benefit from the higher reliability of ZRS storage.
- Move a snapshot to different subscription in the same region for longer retention.

## NOTE

Both subscriptions must be located under the same tenant

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.



[Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure

CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

## Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing <Subscription ID> with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id of the subscription where snapshot exists  
sourceSubscriptionId=<subscriptionId>  
  
#Provide the name of your resource group where snapshot exists  
sourceResourceGroupName=mySourceResourceGroupName  
  
#Provide the name of the snapshot  
snapshotName=mySnapshotName  
  
#Set the context to the subscription Id where snapshot exists  
az account set --subscription $sourceSubscriptionId  
  
#Get the snapshot Id  
snapshotId=$(az snapshot show --name $snapshotName --resource-group $sourceResourceGroupName --query [id] -o tsv)  
  
#If snapshotId is blank then it means that snapshot does not exist.  
echo 'source snapshot Id is: '$snapshotId  
  
#Provide the subscription Id of the subscription where snapshot will be copied to  
#If snapshot is copied to the same subscription then you can skip this step  
targetSubscriptionId=6492b1f7-f219-446b-b509-314e17e1efb0  
  
#Name of the resource group where snapshot will be copied to  
targetResourceGroupName=mytargetResourceGroupName  
  
#Set the context to the subscription Id where snapshot will be copied to  
#If snapshot is copied to the same subscription then you can skip this step  
az account set --subscription $targetSubscriptionId  
  
#Copy snapshot to different subscription using the snapshot Id  
#We recommend you to store your snapshots in Standard storage to reduce cost. Please use Standard_ZRS in  
regions where zone redundant storage (ZRS) is available, otherwise use Standard_LRS  
#Please check out the availability of ZRS here: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-zrs#support-coverage-and-regional-availability  
az snapshot create --resource-group $targetResourceGroupName --name $snapshotName --source $snapshotId --sku Standard_LRS
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name mySourceResourceGroupName
```

## Sample reference

This script uses following commands to create a snapshot in the target subscription using the `Id` of the source snapshot. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az snapshot show</a>	Gets all the properties of a snapshot using the name and resource group properties of the snapshot. The <code>Id</code> property is used to copy the snapshot to different subscription.
<a href="#">az snapshot create</a>	Copies a snapshot by creating a snapshot in different subscription using the <code>Id</code> and name of the parent snapshot.

## Next steps

[Create a virtual machine from a snapshot](#)

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine and managed disks CLI script samples can be found in the [Azure Linux VM documentation](#).

# Create a virtual machine from a snapshot with CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script creates a virtual machine from a snapshot of an OS disk.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription="<subscriptionId>" # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id of the subscription where you want to create Managed Disks
subscriptionId=<subscriptionId>"c

#Provide the name of your resource group
resourceGroupName=myResourceGroupName

#Provide the name of the snapshot that will be used to create Managed Disks
snapshotName=mySnapshotName

#Provide the name of the Managed Disk
osDiskName=myOSDiskName

#Provide the size of the disks in GB. It should be greater than the VHD file size.
diskSize=128

#Provide the storage type for Managed Disk. Premium_LRS or Standard_LRS.
storageType=Premium_LRS

#Provide the OS type
osType=linux

#Provide the name of the virtual machine
virtualMachineName=myVirtualMachineName

#Set the context to the subscription Id where Managed Disk will be created
az account set --subscription $subscriptionId

#Get the snapshot Id
snapshotId=$(az snapshot show --name $snapshotName --resource-group $resourceGroupName --query [id] -o tsv)

#Create a new Managed Disks using the snapshot Id
az disk create --resource-group $resourceGroupName --name $osDiskName --sku $storageType --size-gb $diskSize --source $snapshotId

#Create VM by attaching created managed disks as OS
az vm create --name $virtualMachineName --resource-group $resourceGroupName --attach-os-disk $osDiskName --os-type $osType
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses the following commands to create a managed disk, virtual machine, and all related resources. Each command in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az snapshot show</a>	Gets snapshot using snapshot name and resource group name. Id property of the returned object is used to create a managed disk.
<a href="#">az disk create</a>	Creates managed disks from a snapshot using snapshot Id, disk name, storage type, and size

COMMAND	NOTES
<code>az vm create</code>	Creates a VM using a managed OS disk

## Next steps

For more information on the Azure CLI, see [Azure CLI documentation](#).

Additional virtual machine CLI script samples can be found in the [Azure Linux VM documentation](#).

# Create a virtual machine using an existing managed OS disk with CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

This script creates a virtual machine by attaching an existing managed disk as OS disk. Use this script in preceding scenarios:

- Create a VM from an existing managed OS disk that was copied from a managed disk in different subscription
- Create a VM from an existing managed disk that was created from a specialized VHD file
- Create a VM from an existing managed OS disk that was created from a snapshot

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.



[Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

### Launch Azure Cloud Shell

The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com>.

When Cloud Shell opens, verify that **Bash** is selected for your environment. Subsequent sessions will use Azure CLI in a Bash environment. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press **Enter** to run it.

### Sign in to Azure

Cloud Shell is automatically authenticated under the initial account signed-in with. Use the following script to sign in using a different subscription, replacing `<Subscription ID>` with your Azure Subscription ID. If you don't

have an [Azure subscription](#), create an [Azure free account](#) before you begin.

```
subscription=<subscriptionId> # add subscription here  
az account set -s $subscription # ...or use 'az login'
```

For more information, see [set active subscription](#) or [log in interactively](#)

## Run the script

```
#Provide the subscription Id  
subscriptionId=<subscriptionId>  
  
#Provide the name of your resource group  
resourceGroupName=myResourceGroupName  
  
#Provide the name of the Managed Disk  
managedDiskName=myDiskName  
  
#Provide the OS type  
osType=linux  
  
#Provide the name of the virtual machine  
virtualMachineName=myVirtualMachineName123  
  
#Set the context to the subscription Id where Managed Disk exists and where VM will be created  
az account set --subscription $subscriptionId  
  
#Get the resource Id of the managed disk  
managedDiskId=$(az disk show --name $managedDiskName --resource-group $resourceGroupName --query [id] -o tsv)  
  
#Create VM by attaching existing managed disks as OS  
az vm create --name $virtualMachineName --resource-group $resourceGroupName --attach-os-disk $managedDiskId --os-type $osType
```

## Clean up resources

Run the following command to remove the resource group, VM, and all related resources.

```
az group delete --name myResourceGroupName
```

## Sample reference

This script uses the following commands to get managed disk properties, attach a managed disk to a new VM and create a VM. Each item in the table links to command specific documentation.

COMMAND	NOTES
<a href="#">az disk show</a>	Gets managed disk properties using disk name and resource group name. Id property is used to attach a managed disk to a new VM
<a href="#">az vm create</a>	Creates a VM using a managed OS disk

## Next steps

For more information on the Azure CLI, see [Azure CLI documentation](#).

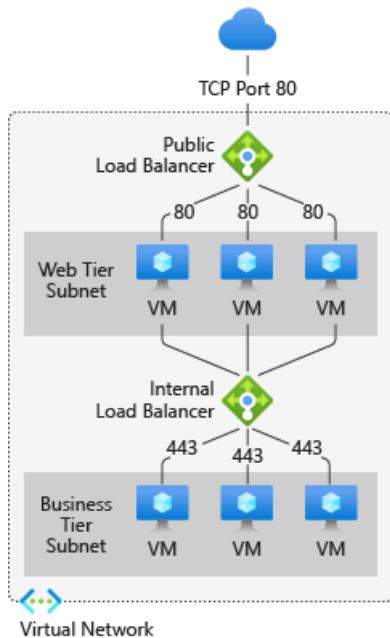
Additional virtual machine CLI script samples can be found in the [Azure Linux VM documentation](#).

# Virtual networks and virtual machines in Azure

9/21/2022 • 17 minutes to read • [Edit Online](#)

When you create a virtual machine (VM), you create a [virtual network](#) or use an existing one. Decide how your virtual machines are intended to be accessed on the virtual network. It's important to [plan before creating resources](#) and make sure you understand the [limits of networking resources](#).

In the following figure, virtual machines are represented as web servers and application servers. Each set of virtual machines is assigned to separate subnets in the virtual network.



You can create a virtual network before you create a virtual machine or you can create the virtual network as you create a virtual machine.

You create these resources to support communication with a virtual machine:

- Network interfaces
- IP addresses
- Virtual network and subnets

Additionally, consider these optional resources:

- Network security groups
- Load balancers

## Network interfaces

A [network interface \(NIC\)](#) is the interconnection between a virtual machine and a virtual network. A virtual machine must have at least one NIC. A virtual machine can have more than one NIC, depending on the size of the VM you create. To learn about the number of NICs each virtual machine size supports, see [VM sizes](#).

You can create a VM with multiple NICs, and add or remove NICs through the lifecycle of a VM. Multiple NICs allow a VM to connect to different subnets.

Each NIC attached to a VM must exist in the same location and subscription as the VM. Each NIC must be connected to a VNet that exists in the same Azure location and subscription as the NIC. You can change the

subnet a VM is connected to after it's created. You can't change the virtual network. Each NIC attached to a VM is assigned a MAC address that doesn't change until the VM is deleted.

This table lists the methods that you can use to create a network interface.

METHOD	DESCRIPTION
Azure portal	When you create a VM in the Azure portal, a network interface is automatically created for you. The portal creates a VM with only one NIC. If you want to create a VM with more than one NIC, you must create it with a different method.
Azure PowerShell	Use <a href="#">New-AzNetworkInterface</a> with the <code>-PublicIpAddressId</code> parameter to provide the identifier of the public IP address that you previously created.
Azure CLI	To provide the identifier of the public IP address that you previously created, use <a href="#">az network nic create</a> with the <code>--public-ip-address</code> parameter.
Template	For information on deploying a networking interface using a template, see <a href="#">Network Interface in a Virtual Network with Public IP Address</a> .

## IP addresses

You can assign these types of [IP addresses](#) to a network interface in Azure:

- **Public IP addresses** - Used to communicate inbound and outbound (without network address translation (NAT)) with the Internet and other Azure resources not connected to a virtual network. Assigning a public IP address to a NIC is optional. Public IP addresses have a nominal charge, and there's a maximum number that can be used per subscription.
- **Private IP addresses** - Used for communication within a virtual network, your on-premises network, and the Internet (with NAT). At least one private IP address must be assigned to a VM. To learn more about NAT in Azure, read [Understanding outbound connections in Azure](#).

You can assign public IP addresses to:

- Virtual machines
- Public load balancers

You can assign private IP address to:

- Virtual machines
- Internal load balancers

You assign IP addresses to a VM using a network interface.

There are two methods in which an IP address is given to a resource, dynamic or static. The default method that Azure gives IP addresses is dynamic. An IP address isn't given when it's created. Instead, the IP address is given when you create a VM or start a stopped VM. The IP address is released when you stop or delete the VM.

To ensure the IP address for the VM remains the same, you can set the allocation method explicitly to static. In this case, an IP address is assigned immediately. It's released only when you delete the VM or change its allocation method to dynamic.

This table lists the methods that you can use to create an IP address.

METHOD	DESCRIPTION
Azure portal	By default, public IP addresses are dynamic. The IP address may change when the VM is stopped or deleted. To guarantee that the VM always uses the same public IP address, create a static public IP address. By default, the portal assigns a dynamic private IP address to a NIC when creating a VM. You can change this IP address to static after the VM is created.
Azure PowerShell	You use <a href="#">New-AzPublicIpAddress</a> with the <code>-AllocationMethod</code> parameter as Dynamic or Static.
Azure CLI	You use <a href="#">az network public-ip create</a> with the <code>--allocation-method</code> parameter as Dynamic or Static.
Template	For more information on deploying a public IP address using a template, see <a href="#">Network Interface in a Virtual Network with Public IP Address</a> .

After you create a public IP address, you can associate it with a VM by assigning it to a NIC.

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Virtual network and subnets

A subnet is a range of IP addresses in the virtual network. You can divide a virtual network into multiple subnets for organization and security. Each NIC in a VM is connected to one subnet in one virtual network. NICs connected to subnets (same or different) within a virtual network can communicate with each other without any extra configuration.

When you set up a virtual network, you specify the topology, including the available address spaces and subnets. Select address ranges that don't overlap if the virtual network is connected to other virtual networks or on-premises networks. The IP addresses are private and can't be accessed from the Internet. Azure treats any address range as part of the private virtual network IP address space. The address range is only reachable within the virtual network, within interconnected virtual networks, and from your on-premises location.

If you work within an organization in which someone else is responsible for the internal networks, talk to that person before selecting your address space. Ensure there's no overlap in the address space. Communicate to them the space you want to use so they don't try to use the same range of IP addresses.

There aren't security boundaries by default between subnets. Virtual machines in each of these subnets can

communicate. If your deployment requires security boundaries, use **Network Security Groups (NSGs)**, which control the traffic flow to and from subnets and to and from VMs.

This table lists the methods that you can use to create a virtual network and subnets.

METHOD	DESCRIPTION
Azure portal	If you let Azure create a virtual network when you create a VM, the name is a combination of the resource group name that contains the virtual network and <code>-vnet</code> . The address space is 10.0.0.0/24, the required subnet name is <b>default</b> , and the subnet address range is 10.0.0.0/24.
Azure PowerShell	You use <a href="#">New-AzVirtualNetworkSubnetConfig</a> and <a href="#">New-AzVirtualNetwork</a> to create a subnet and a virtual network. You can also use <a href="#">Add-AzVirtualNetworkSubnetConfig</a> to add a subnet to an existing virtual network.
Azure CLI	The subnet and the virtual network are created at the same time. Provide a <code>--subnet-name</code> parameter to <a href="#">az network vnet create</a> with the subnet name.
Template	For more information on using a template to create a virtual network and subnets, see <a href="#">Virtual Network with two subnets</a> .

## Network security groups

A **network security group (NSG)** contains a list of Access Control List (ACL) rules that allow or deny network traffic to subnets, NICs, or both. NSGs can be associated with either subnets or individual NICs connected to a subnet. When an NSG is associated with a subnet, the ACL rules apply to all the VMs in that subnet. Traffic to an individual NIC can be restricted by associating an NSG directly to a NIC.

NSGs contain two sets of rules, inbound and outbound. The priority for a rule must be unique within each set.

Each rule has properties of:

- Protocol
- Source and destination port ranges
- Address prefixes
- Direction of traffic
- Priority
- Access type

All NSGs contain a set of default rules. The default rules cannot be deleted. They're assigned the lowest priority and can't be overridden by the rules that you create.

When you associate an NSG to a NIC, the network access rules in the NSG are applied only to that NIC. If an NSG is applied to a single NIC on a multi-NIC VM, it doesn't affect traffic to the other NICs. You can associate different NSGs to a NIC (or VM, depending on the deployment model) and the subnet that a NIC or VM is bound to. Priority is given based on the direction of traffic.

Be sure to [plan](#) your NSGs when you plan your virtual machines and virtual network.

This table lists the methods that you can use to create a network security group.

METHOD	DESCRIPTION
Azure portal	<p>When you create a VM in the Azure portal, an NSG is automatically created and associated to the NIC the portal creates. The name of the NSG is a combination of the name of the VM and <code>-nsg</code>.</p> <p>This NSG contains one inbound rule:  With a priority of 1000.  The service set to RDP.  The protocol set to TCP.  The port set to 3389.  The action set to <b>Allow</b>.  If you want to allow any other inbound traffic to the VM, create another rule or rules.</p>
Azure PowerShell	<p>Use <a href="#">New-AzNetworkSecurityRuleConfig</a> and provide the required rule information. Use <a href="#">New-AzNetworkSecurityGroup</a> to create the NSG. Use <a href="#">Set-AzVirtualNetworkSubnetConfig</a> to configure the NSG for the subnet. Use <a href="#">Set-AzVirtualNetwork</a> to add the NSG to the virtual network.</p>
Azure CLI	<p>Use <a href="#">az network nsg create</a> to initially create the NSG. Use <a href="#">az network nsg rule create</a> to add rules to the NSG. Use <a href="#">az network vnet subnet update</a> to add the NSG to the subnet.</p>
Template	<p>Use <a href="#">Create a Network Security Group</a> as a guide for deploying a network security group using a template.</p>

## Load balancers

[Azure Load Balancer](#) delivers high availability and network performance to your applications. A load balancer can be configured to [balance incoming Internet traffic](#) to VMs or [balance traffic between VMs in a VNet](#). A load balancer can also balance traffic between on-premises computers and VMs in a cross-premises network, or forward external traffic to a specific VM.

The load balancer maps incoming and outgoing traffic between:

- The public IP address and port on the load balancer.
- The private IP address and port of the VM.

When you create a load balancer, you must also consider these configuration elements:

- **Front-end IP configuration** – A load balancer can include one or more front-end IP addresses. These IP addresses serve as ingress for the traffic.
- **Back-end address pool** – IP addresses that are associated with the NIC to which load is distributed.
- **Port Forwarding** - Defines how inbound traffic flows through the front-end IP and distributed to the back-end IP using inbound NAT rules.
- **Load balancer rules** - Maps a given front-end IP and port combination to a set of back-end IP addresses and port combination. A single load balancer can have multiple load-balancing rules. Each rule is a combination of a front-end IP and port and back-end IP and port associated with VMs.
- **Probes** - Monitors the health of VMs. When a probe fails to respond, the load balancer stops sending new connections to the unhealthy VM. The existing connections aren't affected, and new connections are sent to healthy VMs.
- **Outbound rules** - An outbound rule configures outbound Network Address Translation (NAT) for all virtual machines or instances identified by the backend pool of your Standard Load Balancer to be translated to the

frontend.

This table lists the methods that you can use to create an internet-facing load balancer.

METHOD	DESCRIPTION
Azure portal	You can <a href="#">load balance internet traffic to VMs using the Azure portal</a> .
Azure PowerShell	To provide the identifier of the public IP address that you previously created, use <code>New-AzLoadBalancerFrontendIpConfig</code> with the <code>-PublicIpAddress</code> parameter. Use <code>New-AzLoadBalancerBackendAddressPoolConfig</code> to create the configuration of the back-end address pool. Use <code>New-AzLoadBalancerInboundNatRuleConfig</code> to create inbound NAT rules associated with the front-end IP configuration that you created. Use <code>New-AzLoadBalancerProbeConfig</code> to create the probes that you need. Use <code>New-AzLoadBalancerRuleConfig</code> to create the load balancer configuration. Use <code>New-AzLoadBalancer</code> to create the load balancer.
Azure CLI	Use <code>az network lb create</code> to create the initial load balancer configuration. Use <code>az network lb frontend-ip create</code> to add the public IP address that you previously created. Use <code>az network lb address-pool create</code> to add the configuration of the back-end address pool. Use <code>az network lb inbound-nat-rule create</code> to add NAT rules. Use <code>az network lb rule create</code> to add the load balancer rules. Use <code>az network lb probe create</code> to add the probes.
Template	Use <a href="#">3 VMs in a Load Balancer</a> as a guide for deploying a load balancer using a template.

This table lists the methods that you can use to create an internal load balancer.

METHOD	DESCRIPTION
Azure portal	You can <a href="#">balance internal traffic load with a load balancer in the Azure portal</a> .
Azure PowerShell	To provide a private IP address in the network subnet, use <code>New-AzLoadBalancerFrontendIpConfig</code> with the <code>-PrivateIpAddress</code> parameter. Use <code>New-AzLoadBalancerBackendAddressPoolConfig</code> to create the configuration of the back-end address pool. Use <code>New-AzLoadBalancerInboundNatRuleConfig</code> to create inbound NAT rules associated with the front-end IP configuration that you created. Use <code>New-AzLoadBalancerProbeConfig</code> to create the probes that you need. Use <code>New-AzLoadBalancerRuleConfig</code> to create the load balancer configuration. Use <code>New-AzLoadBalancer</code> to create the load balancer.

METHOD	DESCRIPTION
Azure CLI	Use the <a href="#">az network lb create</a> command to create the initial load balancer configuration. To define the private IP address, use <a href="#">az network lb frontend-ip create</a> with the <code>--private-ip-address</code> parameter. Use <a href="#">az network lb address-pool create</a> to add the configuration of the backend address pool. Use <a href="#">az network lb inbound-nat-rule create</a> to add NAT rules. Use <a href="#">az network lb rule create</a> to add the load balancer rules. Use <a href="#">az network lb probe create</a> to add the probes.
Template	Use <a href="#">2 VMs in a Load Balancer</a> as a guide for deploying a load balancer using a template.

## Virtual machines

Virtual machines can be created in the same virtual network and they can connect to each other using private IP addresses. Virtual machines can connect if they're in different subnets. They connect without the need to configure a gateway or use public IP addresses. To put VMs into a virtual network, you create the virtual network. As you create each VM, you assign it to the virtual network and subnet. Virtual machines acquire their network settings during deployment or startup.

Virtual machines are assigned an IP address when they're deployed. When you deploy multiple VMs into a virtual network or subnet, they're assigned IP addresses as they boot up. You can also assign a static IP to a VM. If you assign a static IP, you should consider using a specific subnet to avoid accidentally reusing a static IP for another VM.

If you create a VM and later want to migrate it into a virtual network, it isn't a simple configuration change. Redeploy the VM into the virtual network. The easiest way to redeploy is to delete the VM, but not any disks attached to it, and then re-create the VM using the original disks in the virtual network.

This table lists the methods that you can use to create a VM in a VNet.

METHOD	DESCRIPTION
Azure portal	Uses the default network settings that were previously mentioned to create a VM with a single NIC. To create a VM with multiple NICs, you must use a different method.
Azure PowerShell	Includes the use of <a href="#">Add-AzVMNetworkInterface</a> to add the NIC that you previously created to the VM configuration.
Azure CLI	Create and connect a VM to a virtual network, subnet, and NIC that builds as individual steps.
Template	Use <a href="#">Very simple deployment of a Windows VM</a> as a guide for deploying a VM using a template.

## Virtual network NAT

Virtual Network NAT (network address translation) simplifies outbound-only Internet connectivity for virtual networks. When configured on a subnet, all outbound connectivity uses your specified static public IP addresses. Outbound connectivity is possible without load balancer or public IP addresses directly attached to virtual machines. NAT is fully managed and highly resilient.

Outbound connectivity can be defined for each subnet with NAT. Multiple subnets within the same virtual network can have different NATs. A subnet is configured by specifying which NAT gateway resource to use. All UDP and TCP outbound flows from any virtual machine instance will use NAT. NAT is compatible with standard SKU public IP address resources or public IP prefix resources or a combination of both. You can use a public IP prefix directly or distribute the public IP addresses of the prefix across multiple NAT gateway resources. NAT will groom all traffic to the range of IP addresses of the prefix. Any IP filtering of your deployments is easier.

All outbound traffic for the subnet is processed by NAT automatically without any customer configuration. User-defined routes aren't necessary. NAT takes precedence over other outbound scenarios and replaces the default Internet destination of a subnet.

Virtual machines created by Virtual machine scale sets Flexible Orchestration mode don't have default outbound access. Virtual network NAT is the recommended outbound access method for Virtual machine scale sets Flexible Orchestration Mode.

For more information about the NAT gateway resource and virtual network NAT, see [What is Azure Virtual Network NAT?](#).

This table lists the methods that you can use to create a NAT gateway resource.

METHOD	DESCRIPTION
<a href="#">Azure portal</a>	Creates a virtual network, subnet, public IP, NAT gateway, and a virtual machine to test the NAT gateway resource.
<a href="#">Azure PowerShell</a>	Includes the use of <a href="#">New-AzNatGateway</a> to create a NAT gateway resource. Creates a virtual network, subnet, public IP, NAT gateway, and a virtual machine to test the NAT gateway resource.
<a href="#">Azure CLI</a>	Includes the use of <a href="#">az network nat gateway create</a> to create a NAT gateway resource. Creates a virtual network, subnet, public IP, NAT gateway, and a virtual machine to test the NAT gateway resource.
<a href="#">Template</a>	Creates a virtual network, subnet, public IP, and NAT gateway resource.

## Azure Bastion

Azure Bastion is deployed to provide secure management connectivity to virtual machines in a virtual network. Azure Bastion Service enables you to securely and seamlessly RDP & SSH to the VMs in your virtual network. Azure bastion enables connections without exposing a public IP on the VM. Connections are made directly from the Azure portal, without the need of an extra client/agent or piece of software. Azure Bastion supports standard SKU public IP addresses.

For more information about Azure Bastion, see [What is Azure Bastion?](#).

This table lists the methods you can use to create an Azure Bastion deployment.

METHOD	DESCRIPTION
<a href="#">Azure portal</a>	Creates a virtual network, subnets, public IP, bastion host, and virtual machines.

METHOD	DESCRIPTION
Azure PowerShell	Creates a virtual network, subnets, public IP, and bastion host. Includes the use of <a href="#">New-AzBastion</a> to create the bastion host.
Azure CLI	Creates a virtual network, subnets, public IP, and bastion host. Includes the use of <a href="#">az network bastion create</a> to create the bastion host.
Template	For an example of a template deployment that integrates an Azure Bastion host with a sample deployment, see <a href="#">Quickstart: Create a public load balancer to load balance VMs by using an ARM template</a> .

## Next steps

For VM-specific steps on how to manage Azure virtual networks for VMs, see the [Windows](#) or [Linux](#) tutorials.

There are also tutorials on how to load balance VMs and create highly available applications for [Windows](#) or [Linux](#).

- Learn how to configure [user-defined routes and IP forwarding](#).
- Learn how to configure [VNet to VNet connections](#).
- Learn how to [Troubleshoot routes](#).
- Learn more about [Virtual machine network bandwidth](#).

# Optimize network throughput for Azure virtual machines

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure virtual machines (VM) have default network settings that can be further optimized for network throughput. This article describes how to optimize network throughput for Microsoft Azure Windows and Linux VMs, including major distributions such as Ubuntu, CentOS, and Red Hat.

## Windows VM

If your Windows VM supports [Accelerated Networking](#), enabling that feature would be the optimal configuration for throughput. For all other Windows VMs, using Receive Side Scaling (RSS) can reach higher maximal throughput than a VM without RSS. RSS may be disabled by default in a Windows VM. To determine whether RSS is enabled, and enable it if it's currently disabled, complete the following steps:

1. See if RSS is enabled for a network adapter with the `Get-NetAdapterRss` PowerShell command. In the following example output returned from the `Get-NetAdapterRss`, RSS is not enabled.

```
Name          : Ethernet
InterfaceDescription : Microsoft Hyper-V Network Adapter
Enabled       : False
```

2. To enable RSS, enter the following command:

```
Get-NetAdapter | % {Enable-NetAdapterRss -Name $_.Name}
```

The previous command does not have an output. The command changed NIC settings, causing temporary connectivity loss for about one minute. A Reconnecting dialog box appears during the connectivity loss. Connectivity is typically restored after the third attempt.

3. Confirm that RSS is enabled in the VM by entering the `Get-NetAdapterRss` command again. If successful, the following example output is returned:

```
Name      : Ethernet
InterfaceDescription : Microsoft Hyper-V Network Adapter
Enabled    : True
```

## Linux VM

RSS is always enabled by default in an Azure Linux VM. Linux kernels released since October 2017 include new network optimizations options that enable a Linux VM to achieve higher network throughput.

### Ubuntu for new deployments

The Ubuntu Azure kernel is the most optimized for network performance on Azure. To get the latest optimizations, first install the latest supported version of 18.04-LTS, as follows:

```
"Publisher": "Canonical",
"Offer": "UbuntuServer",
"Sku": "18.04-LTS",
"Version": "latest"
```

After the creation is complete, enter the following commands to get the latest updates. These steps also work for VMs currently running the Ubuntu Azure kernel.

```
#run as root or preface with sudo
apt-get -y update
apt-get -y upgrade
apt-get -y dist-upgrade
```

The following optional command set may be helpful for existing Ubuntu deployments that already have the Azure kernel but that have failed to further updates with errors.

```
#optional steps may be helpful in existing deployments with the Azure kernel
#run as root or preface with sudo
apt-get -f install
apt-get --fix-missing install
apt-get clean
apt-get -y update
apt-get -y upgrade
apt-get -y dist-upgrade
```

#### **Ubuntu Azure kernel upgrade for existing VMs**

Significant throughput performance can be achieved by upgrading to the Azure Linux kernel. To verify whether you have this kernel, check your kernel version. It should be the same or later than the example.

```
#Azure kernel name ends with "-azure"
uname -r

#sample output on Azure kernel:
#4.13.0-1007-azure
```

If your VM does not have the Azure kernel, the version number usually begins with "4.4." If the VM does not have the Azure kernel, run the following commands as root:

```
#run as root or preface with sudo
apt-get update
apt-get upgrade -y
apt-get dist-upgrade -y
apt-get install "linux-azure"
reboot
```

#### **CentOS**

In order to get the latest optimizations, it is best to create a VM with the latest supported version by specifying the following parameters:

```
"Publisher": "OpenLogic",
"Offer": "Centos",
"Sku": "7.7",
"Version": "latest"
```

New and existing VMs can benefit from installing the latest Linux Integration Services (LIS). The throughput

optimization is in LIS, starting from 4.2.2-2, although later versions contain further improvements. Enter the following commands to install the latest LIS:

```
sudo yum update  
sudo reboot  
sudo yum install microsoft-hyper-v
```

## Red Hat

In order to get the optimizations, it is best to create a VM with the latest supported version by specifying the following parameters:

```
"Publisher": "RedHat"  
"Offer": "RHEL"  
"Sku": "7-RAW"  
"Version": "latest"
```

New and existing VMs can benefit from installing the latest Linux Integration Services (LIS). The throughput optimization is in LIS, starting from 4.2. Enter the following commands to download and install LIS:

```
wget https://aka.ms/lis  
tar xvf lis  
cd LISISO  
sudo ./install.sh #or upgrade.sh if prior LIS was previously installed
```

Learn more about Linux Integration Services Version 4.2 for Hyper-V by viewing the [download page](#).

## Next steps

- Deploy VMs close to each other for low latency with [Proximity Placement Group](#)
- See the optimized result with [Bandwidth/Throughput testing Azure VM](#) for your scenario.
- Read about how [bandwidth is allocated to virtual machines](#)
- Learn more with [Azure Virtual Network frequently asked questions \(FAQ\)](#)

# Azure Virtual Network concepts and best practices

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article describes key concepts and best practices for Azure Virtual Network (VNet).

## VNet concepts

- **Address space:** When creating a VNet, you must specify a custom private IP address space using public and private (RFC 1918) addresses. Azure assigns resources in a virtual network a private IP address from the address space that you assign. For example, if you deploy a VM in a VNet with address space, 10.0.0.0/16, the VM will be assigned a private IP like 10.0.0.4.
- **Subnets:** Subnets enable you to segment the virtual network into one or more sub-networks and allocate a portion of the virtual network's address space to each subnet. You can then deploy Azure resources in a specific subnet. Just like in a traditional network, subnets allow you to segment your VNet address space into segments that are appropriate for the organization's internal network. This also improves address allocation efficiency. You can secure resources within subnets using Network Security Groups. For more information, see [Network security groups](#).
- **Regions:** VNet is scoped to a single region/location; however, multiple virtual networks from different regions can be connected together using Virtual Network Peering.
- **Subscription:** VNet is scoped to a subscription. You can implement multiple virtual networks within each Azure [subscription](#) and Azure [region](#).

## Best practices

As you build your network in Azure, it is important to keep in mind the following universal design principles:

- Ensure non-overlapping address spaces. Make sure your VNet address space (CIDR block) does not overlap with your organization's other network ranges.
- Your subnets should not cover the entire address space of the VNet. Plan ahead and reserve some address space for the future.
- It is recommended you have fewer large VNets rather than multiple small VNets. This will prevent management overhead.
- Secure your VNets by assigning Network Security Groups (NSGs) to the subnets beneath them. For more information about network security concepts, see [Azure network security overview](#).

## Next steps

To get started using a virtual network, create one, deploy a few VMs to it, and communicate between the VMs. To learn how, see the [Create a virtual network](#) quickstart.

# Quickstart: Create a virtual network using the Azure CLI

9/21/2022 • 4 minutes to read • [Edit Online](#)

A virtual network enables Azure resources, like virtual machines (VMs), to communicate privately with each other, and with the internet.

In this quickstart, you learn how to create a virtual network. After creating a virtual network, you deploy two VMs into the virtual network. You then connect to the VMs from the internet, and communicate privately over the new virtual network.

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This quickstart requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group and a virtual network

Before you can create a virtual network, you have to create a resource group to host the virtual network. Create a resource group with [az group create](#). This example creates a resource group named `CreateVNetQS-rg` in the `Eastus` location:

```
az group create \
  --name CreateVNetQS-rg \
  --location eastus
```

Create a virtual network with [az network vnet create](#). This example creates a default virtual network named `myVNet` with one subnet named `default`:

```
az network vnet create \
--name myVNet \
--resource-group CreateVNetQS-rg \
--subnet-name default
```

## Create virtual machines

Create two VMs in the virtual network.

### Create the first VM

Create a VM with [az vm create](#).

If SSH keys don't already exist in a default key location, the command creates them. To use a specific set of keys, use the `--ssh-key-value` option.

The `--no-wait` option creates the VM in the background. You can continue to the next step.

This example creates a VM named **myVM1**:

```
az vm create \
--resource-group CreateVNetQS-rg \
--name myVM1 \
--image UbuntuLTS \
--generate-ssh-keys \
--public-ip-address myPublicIP-myVM1 \
--no-wait
```

### Create the second VM

You used the `--no-wait` option in the previous step. You can go ahead and create the second VM named **myVM2**.

```
az vm create \
--resource-group CreateVNetQS-rg \
--name myVM2 \
--image UbuntuLTS \
--public-ip-address myPublicIP-myVM2 \
--generate-ssh-keys
```

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

#### Azure CLI output message

The VMs take a few minutes to create. After Azure creates the VMs, the Azure CLI returns output like this:

```
{  
  "fqdns": "",  
  "id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/CreateVNetQS-  
rg/providers/Microsoft.Compute/virtualMachines/myVM2",  
  "location": "eastus",  
  "macAddress": "00-0D-3A-23-9A-49",  
  "powerState": "VM running",  
  "privateIpAddress": "10.0.0.5",  
  "publicIpAddress": "40.68.254.142",  
  "resourceGroup": "CreateVNetQS-rg"  
  "zones": ""  
}
```

## VM public IP

To get the public IP address myVM2, use [az network public-ip show](#):

```
az network public-ip show \  
  --resource-group CreateVNetQS-rg \  
  --name myPublicIP-myVM2 \  
  --query ipAddress \  
  --output tsv
```

## Connect to a VM from the internet

In this command, replace <publicIpAddress> with the public IP address of your myVM2 VM:

```
ssh <publicIpAddress>
```

## Communicate between VMs

To confirm private communication between the myVM2 and myVM1 VMs, enter [ping myVM1 -c 4](#).

You'll receive a reply message like this:

```
azureuser@myVM2:~$ ping myVM1 -c 4  
PING myVM1.h0o2foz2r0tefnccdcnfqm2lid.bx.internal.cloudapp.net (10.0.0.4) 56(84) bytes of data.  
64 bytes from myvm1.internal.cloudapp.net (10.0.0.4): icmp_seq=1 ttl=64 time=2.77 ms  
64 bytes from myvm1.internal.cloudapp.net (10.0.0.4): icmp_seq=2 ttl=64 time=1.95 ms  
64 bytes from myvm1.internal.cloudapp.net (10.0.0.4): icmp_seq=3 ttl=64 time=2.19 ms  
64 bytes from myvm1.internal.cloudapp.net (10.0.0.4): icmp_seq=4 ttl=64 time=1.85 ms  
  
--- myVM1.h0o2foz2r0tefnccdcnfqm2lid.bx.internal.cloudapp.net ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3003ms  
rtt min/avg/max/mdev = 1.859/2.195/2.770/0.357 ms
```

Exit the SSH session with the myVM2 VM.

## Clean up resources

When no longer needed, you can use [az group delete](#) to remove the resource group and all the resources it has:

```
az group delete \
--name CreateVNetQS-rg \
--yes
```

## Next steps

In this quickstart:

- You created a default virtual network and two VMs.
- You connected to one VM from the internet and communicated privately between the two VMs.

Private communication between VMs is unrestricted in a virtual network.

Advance to the next article to learn more about configuring different types of VM network communications:

[Filter network traffic](#)

# Tutorial: Filter network traffic with a network security group using the Azure portal

9/21/2022 • 8 minutes to read • [Edit Online](#)

You can use a network security group to filter inbound and outbound network traffic to and from Azure resources in an Azure virtual network.

Network security groups contain security rules that filter network traffic by IP address, port, and protocol. When a network security group is associated with a subnet, security rules are applied to resources deployed in that subnet.

In this tutorial, you learn how to:

- Create a network security group and security rules
- Create application security groups
- Create a virtual network and associate a network security group to a subnet
- Deploy virtual machines and associate their network interfaces to the application security groups
- Test traffic filters

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Prerequisites

- An Azure subscription

## Sign in to Azure

Sign in to the [Azure portal](#).

## Create a virtual network

1. From the Azure portal menu, select + **Create a resource** > **Networking** > **Virtual network**, or search for *Virtual Network* in the portal search box.
2. Select **Create**.
3. On the **Basics** tab of **Create virtual network**, enter or select this information:

SETTING	VALUE
Project details	
Subscription	Select your subscription.
Resource group	Select <b>Create new</b> . Enter <i>myResourceGroup</i> . Select <b>OK</b> .
Instance details	

SETTING	VALUE
Name	Enter <i>myVNet</i> .
Region	Select <b>East US</b> .

4. Select the **Review + create** tab, or select the blue **Review + create** button at the bottom of the page.
5. Select **Create**.

## Create application security groups

An [application security group \(ASGs\)](#) enables you to group together servers with similar functions, such as web servers.

1. From the Azure portal menu, select **+ Create a resource > Networking > Application security group**, or search for *Application security group* in the portal search box.
2. Select **Create**.
3. On the **Basics** tab of **Create an application security group**, enter or select this information:

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>myResourceGroup</b> .
<b>Instance details</b>	
Name	Enter <i>myAsgWebServers</i> .
Region	Select <b>(US) East US</b> .

4. Select the **Review + create** tab, or select the blue **Review + create** button at the bottom of the page.
5. Select **Create**.
6. Repeat the previous steps, specifying the following values:

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>myResourceGroup</b> .
<b>Instance details</b>	
Name	Enter <i>myAsgMgmtServers</i> .
Region	Select <b>(US) East US</b> .

7. Select the **Review + create** tab, or select the blue **Review + create** button at the bottom of the page.

8. Select **Create**.

## Create a network security group

A [network security group \(NSG\)](#) secures network traffic in your virtual network.

1. From the Azure portal menu, select **+ Create a resource > Networking > Network security group**, or search for *Network security group* in the portal search box.

2. Select **Create**.

3. On the **Basics** tab of **Create network security group**, enter or select this information:

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>myResourceGroup</b> .
<b>Instance details</b>	
Name	Enter <i>myNSG</i> .
Location	Select <b>(US) East US</b> .

4. Select the **Review + create** tab, or select the blue **Review + create** button at the bottom of the page.

5. Select **Create**.

## Associate network security group to subnet

In this section, you'll associate the network security group with the subnet of the virtual network you created earlier.

1. Search for *myNsg* in the portal search box.

2. Select **Subnets** from the **Settings** section of **myNSG**.

3. In the **Subnets** page, select **+ Associate**:

**myNSG | Subnets**

Network security group

+ Associate

Name	Address range	Virtual network
No results.		

Search (Cmd+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

- Inbound security rules
- Outbound security rules
- Network interfaces
- Subnets** (highlighted with a red box)
- Properties
- Locks

4. Under **Associate subnet**, select myVNet for **Virtual network**.

5. Select **default** for **Subnet**, and then select **OK**.

## Create security rules

1. Select **Inbound security rules** from the **Settings** section of myNSG.

2. In **Inbound security rules** page, select **+ Add**:

**myNSG | Inbound security rules**

Network security group

+ Add

Priority ↑	Name ↑	Port ↑↓	Protocol ↑↓	Source ↑↓	Destination ↑↓	Action ↑↓
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

Search (Cmd+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

- Inbound security rules** (highlighted with a red box)
- Outbound security rules
- Network interfaces

3. Create a security rule that allows ports 80 and 443 to the **myAsgWebServers** application security group. In **Add inbound security rule** page, enter or select this information:

SETTING	VALUE
Source	Leave the default of <b>Any</b> .
Source port ranges	Leave the default of <b>(*)</b> .
Destination	Select <b>Application security group</b> .
Destination application security groups	Select <b>myAsgWebServers</b> .

SETTING	VALUE
Service	Leave the default of <b>Custom</b> .
Destination port ranges	Enter <b>80,443</b> .
Protocol	Select <b>TCP</b> .
Action	Leave the default of <b>Allow</b> .
Priority	Leave the default of <b>100</b> .
Name	Enter <b>Allow-Web-All</b> .

 Add inbound security rule ×

myNSG

---

Source ⓘ

Source port ranges \* ⓘ

Destination ⓘ

Destination application security group \* ⓘ

Service ⓘ

Destination port ranges \* ⓘ

Protocol  
 Any  
 TCP    
 UDP  
 ICMP

Action  
 Allow  
 Deny

Priority \* ⓘ

Name \*

Description

---

Add Cancel

4. Select **Add**.

5. Complete steps 3-4 again using this information:

SETTING	VALUE
Source	Leave the default of Any.
Source port ranges	Leave the default of (*).
Destination	Select <b>Application security group</b> .
Destination application security group	Select <b>myAsgMgmtServers</b> .
Service	Leave the default of <b>Custom</b> .
Destination port ranges	Enter <b>3389</b> .
Protocol	Select <b>Any</b> .
Action	Leave the default of <b>Allow</b> .
Priority	Leave the default of <b>110</b> .
Name	Enter <b>Allow-RDP-All</b> .

6. Select **Add**.

**Caution**

In this article, RDP (port 3389) is exposed to the internet for the VM that is assigned to the **myAsgMgmtServers** application security group.

For production environments, instead of exposing port 3389 to the internet, it's recommended that you connect to Azure resources that you want to manage using a VPN, private network connection, or Azure Bastion.

For more information on Azure Bastion, see [What is Azure Bastion?](#).

Once you've completed steps 1-3, review the rules you created. Your list should look like the list in the following example:

Priority	Name	Port	Protocol	Source	Destination	Action
100	Allow-Web-All	80,443	TCP	Any	myAsgWebServers	Allow
110	Allow-RDP-All	3389	Any	Any	myAsgMgmtServers	Allow
65000	AllowVnetInBound	Any	Any	VirtualNetwork	VirtualNetwork	Allow
65001	AllowAzureLoadBalancerInBound	Any	Any	AzureLoadBalancer	Any	Allow
65500	DenyAllInBound	Any	Any	Any	Any	Deny

# Create virtual machines

Create two virtual machines (VMs) in the virtual network.

## Create the first virtual machine

1. From the Azure portal menu, select + **Create a resource** > **Compute** > **Virtual machine**, or search for *Virtual machine* in the portal search box.
2. In **Create a virtual machine**, enter or select this information in the **Basics** tab:

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>myResourceGroup</b> .
<b>Instance details</b>	
Virtual machine name	Enter <i>myVMWeb</i> .
Region	Select <b>(US) East US</b> .
Availability options	Leave the default of <b>No infrastructure redundancy required</b> .
Security type	Leave the default of <b>Standard</b> .
Image	Select <b>Windows Server 2019 Datacenter - Gen2</b> .
Azure Spot instance	Leave the default of unchecked.
Size	Select <b>Standard_D2s_V3</b> .
<b>Administrator account</b>	
Username	Enter a username.
Password	Enter a password.
Confirm password	Reenter password.
<b>Inbound port rules</b>	
Select inbound ports	Select <b>None</b> .

3. Select the **Networking** tab.
4. In the **Networking** tab, enter or select the following information:

SETTING	VALUE
<b>Network interface</b>	

SETTING	VALUE
Virtual network	Select myVNet.
Subnet	Select default (10.0.0.0/24).
Public IP	Leave the default of a new public IP.
NIC network security group	Select None.

5. Select the **Review + create** tab, or select the blue **Review + create** button at the bottom of the page.

6. Select **Create**. The VM may take a few minutes to deploy.

### Create the second virtual machine

Complete steps 1-6 again, but in step 2, enter *myVMMgmt* for Virtual machine name.

Wait for the VMs to complete deployment before advancing to the next section.

## Associate network interfaces to an ASG

When you created the VMs, Azure created a network interface for each VM, and attached it to the VM.

Add the network interface of each VM to one of the application security groups you created previously:

1. Search for *myVMWeb* in the portal search box.
2. Select **Networking** from the **Settings** section of **myVMWeb** VM.
3. Select the **Application security groups** tab, then select **Configure the application security groups**.

The screenshot shows the Azure portal interface for managing a virtual machine named 'myVMWeb'. The left sidebar has a 'Settings' section with a 'Networking' item highlighted by a red box. The main content area is titled 'myVMWeb | Networking' and shows a list of network interfaces. One interface, 'myvmweb829', is selected and its details are shown: IP configuration is 'ipconfig1 (Primary)'. Below the interface list, there are tabs for 'Inbound port rules', 'Outbound port rules', 'Application security groups' (which is highlighted with a red box), and 'Load balancing'. A button labeled 'Configure the application security groups' is also highlighted with a red box at the bottom of the interface details.

4. In **Configure the application security groups**, select **myAsgWebServers**. Select **Save**.

Configure the application security groups

myvmweb829

 Save  Discard

 Showing only application security groups in the same region as the network interface. If you choose more than one application security group, they must all exist in the same virtual network.

Application security groups

myAsgWebServers

Filter the application security groups

myresourcegroup

myAsgMgmtServers

myAsgWebServers

5. Complete steps 1 and 2 again, searching for the *myVMMgmt* virtual machine and selecting the **myAsgMgmtServers** ASG.

## Test traffic filters

1. Search for *myVMMgmt* in the portal search box.
2. On the **Overview** page, select the **Connect** button and then select **RDP**.
3. Select **Download RDP file**.
4. Open the downloaded rdp file and select **Connect**. Enter the username and password you specified when creating the VM.
5. Select **OK**.
6. You may receive a certificate warning during the connection process. If you receive the warning, select **Yes or Continue**, to continue with the connection.

The connection succeeds, because inbound traffic from the internet to the **myAsgMgmtServers** application security group is allowed through port 3389.

The network interface for **myVMMgmt** is associated with the **myAsgMgmtServers** application security group and allows the connection.

7. Open a PowerShell session on **myVMMgmt**. Connect to **myVMWeb** using the following:

```
mstsc /v:myVmWeb
```

The RDP connection from **myVMMgmt** to **myVMWeb** succeeds because virtual machines in the same network can communicate with each other over any port by default.

You can't create an RDP connection to the **myVMWeb** virtual machine from the internet. The security rule for the **myAsgWebServers** prevents connections to port 3389 inbound from the internet. Inbound traffic from the Internet is denied to all resources by default.

8. To install Microsoft IIS on the **myVMWeb** virtual machine, enter the following command from a

PowerShell session on the **myVMWeb** virtual machine:

```
Install-WindowsFeature -name Web-Server -IncludeManagementTools
```

9. After the IIS installation is complete, disconnect from the **myVMWeb** virtual machine, which leaves you in the **myVMMgmt** virtual machine remote desktop connection.
10. Disconnect from the **myVMMgmt** VM.
11. Search for *myVMWeb* in the portal search box.
12. On the **Overview** page of **myVMWeb**, note the **Public IP address** for your VM. The address shown in the following example is 23.96.39.113, but your address is different:

The screenshot shows the Azure portal interface. In the top left, there's a breadcrumb navigation: Home >. Below it, a search bar contains 'Search (Cmd+/' followed by a dropdown arrow. To the right of the search bar are several action buttons: Connect, Start, Restart, Stop, Capture, Delete, Refresh, and Open in mobile. The main content area is titled 'myVMWeb' and 'Virtual machine'. On the left, a sidebar menu includes: Overview (selected), Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Networking, Connect, Windows Admin Center (previous), Disks, and Size. The main panel displays the VM details under the 'Essentials' section. Key information includes:

- Resource group: myResourceGroup
- Status: Running
- Location: East US
- Subscription: (change)
- Tags: (change) Click here to add tags
- Operating system: Windows (Windows Server 2019 Datacenter)
- Size: Standard D2s v3 (2 vcpus, 8 GiB memory)
- Public IP address: **23.96.39.113** (highlighted with a red box)
- Virtual network/subnet: myVNet/default
- DNS name: Configure

13. To confirm that you can access the **myVMWeb** web server from the internet, open an internet browser on your computer and browse to `http://<public-ip-address-from-previous-step>`.

You see the IIS default page, because inbound traffic from the internet to the **myAsgWebServers** application security group is allowed through port 80.

The network interface attached for **myVMWeb** is associated with the **myAsgWebServers** application security group and allows the connection.

## Clean up resources

When no longer needed, delete the resource group and all of the resources it contains:

1. Enter *myResourceGroup* in the **Search** box at the top of the portal. When you see **myResourceGroup** in the search results, select it.
2. Select **Delete resource group**.
3. Enter **myResourceGroup** for **TYPE THE RESOURCE GROUP NAME:** and select **Delete**.

## Next steps

In this tutorial, you:

- Created a network security group and associated it to a virtual network subnet.
- Created application security groups for web and management.
- Created two virtual machines and associated their network interfaces with the application security groups.
- Tested the application security group network filtering.

To learn more about network security groups, see [Network security group overview](#) and [Manage a network security group](#).

Azure routes traffic between subnets by default. You may instead, choose to route traffic between subnets through a VM, serving as a firewall, for example.

To learn how to create a route table, advance to the next tutorial.

[Create a route table](#)

# Filter network traffic with a network security group using PowerShell

9/21/2022 • 8 minutes to read • [Edit Online](#)

You can filter network traffic inbound to and outbound from a virtual network subnet with a network security group. Network security groups contain security rules that filter network traffic by IP address, port, and protocol. Security rules are applied to resources deployed in a subnet. In this article, you learn how to:

- Create a network security group and security rules
- Create a virtual network and associate a network security group to a subnet
- Deploy virtual machines (VM) into a subnet
- Test traffic filters

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Azure Cloud Shell

Azure hosts Azure Cloud Shell, an interactive shell environment that you can use through your browser. You can use either Bash or PowerShell with Cloud Shell to work with Azure services. You can use the Cloud Shell preinstalled commands to run the code in this article, without having to install anything on your local environment.

To start Azure Cloud Shell:

OPTION	EXAMPLE/LINK
Select <b>Try It</b> in the upper-right corner of a code or command block. Selecting <b>Try It</b> doesn't automatically copy the code or command to Cloud Shell.	
Go to <a href="https://shell.azure.com">https://shell.azure.com</a> , or select the <b>Launch Cloud Shell</b> button to open Cloud Shell in your browser.	
Select the <b>Cloud Shell</b> button on the menu bar at the upper right in the <a href="#">Azure portal</a> .	

To use Azure Cloud Shell:

1. Start Cloud Shell.
2. Select the **Copy** button on a code block (or command block) to copy the code or command.
3. Paste the code or command into the Cloud Shell session by selecting **Ctrl+Shift+V** on Windows and Linux, or by selecting **Cmd+Shift+V** on macOS.
4. Select **Enter** to run the code or command.

If you choose to install and use PowerShell locally, this article requires the Azure PowerShell module version 1.0.0 or later. Run `Get-Module -ListAvailable Az` to find the installed version. If you need to upgrade, see [Install Azure PowerShell module](#). If you are running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

# Create a network security group

A network security group contains security rules. Security rules specify a source and destination. Sources and destinations can be application security groups.

## Create application security groups

First create a resource group for all the resources created in this article with [New-AzResourceGroup](#). The following example creates a resource group in the *eastus* location:

```
New-AzResourceGroup -ResourceGroupName myResourceGroup -Location EastUS
```

Create an application security group with [New-AzApplicationSecurityGroup](#). An application security group enables you to group servers with similar port filtering requirements. The following example creates two application security groups.

```
$webAsg = New-AzApplicationSecurityGroup ` -ResourceGroupName myResourceGroup ` -Name myAsgWebServers ` -Location eastus

$mgmtAsg = New-AzApplicationSecurityGroup ` -ResourceGroupName myResourceGroup ` -Name myAsgMgmtServers ` -Location eastus
```

## Create security rules

Create a security rule with [New-AzNetworkSecurityRuleConfig](#). The following example creates a rule that allows traffic inbound from the internet to the *myWebServers* application security group over ports 80 and 443:

```
$webRule = New-AzNetworkSecurityRuleConfig ` -Name "Allow-Web-All" ` -Access Allow ` -Protocol Tcp ` -Direction Inbound ` -Priority 100 ` -SourceAddressPrefix Internet ` -SourcePortRange * ` -DestinationApplicationSecurityGroupId $webAsg.id ` -DestinationPortRange 80,443
```

The following example creates a rule that allows traffic inbound from the internet to the *\*myMgmtServers\** application security group over port 3389:

```
$mgmtRule = New-AzNetworkSecurityRuleConfig ` -Name "Allow-RDP-All" ` -Access Allow ` -Protocol Tcp ` -Direction Inbound ` -Priority 110 ` -SourceAddressPrefix Internet ` -SourcePortRange * ` -DestinationApplicationSecurityGroupId $mgmtAsg.id ` -DestinationPortRange 3389
```

In this article, RDP (port 3389) is exposed to the internet for the *myAsgMgmtServers* VM. For production environments, instead of exposing port 3389 to the internet, it's recommended that you connect to Azure resources that you want to manage using a [VPN](#) or [private](#) network connection.

## Create a network security group

Create a network security group with [New-AzNetworkSecurityGroup](#). The following example creates a network security group named *myNsg*:

```
$nsg = New-AzNetworkSecurityGroup `  
    -ResourceGroupName myResourceGroup `  
    -Location eastus `  
    -Name myNsg `  
    -SecurityRules $webRule,$mgmtRule
```

## Create a virtual network

Create a virtual network with [New-AzVirtualNetwork](#). The following example creates a virtual named *myVirtualNetwork*:

```
$virtualNetwork = New-AzVirtualNetwork `  
    -ResourceGroupName myResourceGroup `  
    -Location EastUS `  
    -Name myVirtualNetwork `  
    -AddressPrefix 10.0.0.0/16
```

Create a subnet configuration with [New-AzVirtualNetworkSubnetConfig](#), and then write the subnet configuration to the virtual network with [Set-AzVirtualNetwork](#). The following example adds a subnet named *mySubnet* to the virtual network and associates the *myNsg* network security group to it:

```
Add-AzVirtualNetworkSubnetConfig `  
    -Name mySubnet `  
    -VirtualNetwork $virtualNetwork `  
    -AddressPrefix "10.0.2.0/24" `  
    -NetworkSecurityGroup $nsg  
$virtualNetwork | Set-AzVirtualNetwork
```

## Create virtual machines

Before creating the VMs, retrieve the virtual network object with the subnet with [Get-AzVirtualNetwork](#):

```
$virtualNetwork = Get-AzVirtualNetwork `  
    -Name myVirtualNetwork `  
    -Resourcegroupname myResourceGroup
```

Create a public IP address for each VM with [New-AzPublicIpAddress](#):

```
$publicIpWeb = New-AzPublicIpAddress `  
    -AllocationMethod Dynamic `  
    -ResourceGroupName myResourceGroup `  
    -Location eastus `  
    -Name myVmWeb  
  
$publicIpMgmt = New-AzPublicIpAddress `  
    -AllocationMethod Dynamic `  
    -ResourceGroupName myResourceGroup `  
    -Location eastus `  
    -Name myVmMgmt
```

Create two network interfaces with [New-AzNetworkInterface](#), and assign a public IP address to the network interface. The following example creates a network interface, associates the *myVmWeb* public IP address to it,

and makes it a member of the *myAsgWebServers* application security group:

```
$webNic = New-AzNetworkInterface `  
    -Location eastus `  
    -Name myVmWeb `  
    -ResourceGroupName myResourceGroup `  
    -SubnetId $virtualNetwork.Subnets[0].Id `  
    -ApplicationSecurityGroupId $webAsg.Id `  
    -PublicIpAddressId $publicIpWeb.Id
```

The following example creates a network interface, associates the *myVmMgmt* public IP address to it, and makes it a member of the *myAsgMgmtServers* application security group:

```
$mgmtNic = New-AzNetworkInterface `  
    -Location eastus `  
    -Name myVmMgmt `  
    -ResourceGroupName myResourceGroup `  
    -SubnetId $virtualNetwork.Subnets[0].Id `  
    -ApplicationSecurityGroupId $mgmtAsg.Id `  
    -PublicIpAddressId $publicIpMgmt.Id
```

Create two VMs in the virtual network so you can validate traffic filtering in a later step.

Create a VM configuration with [New-AzVMConfig](#), then create the VM with [New-AzVM](#). The following example creates a VM that will serve as a web server. The `-AsJob` option creates the VM in the background, so you can continue to the next step:

```
# Create user object  
$cred = Get-Credential -Message "Enter a username and password for the virtual machine."  
  
$webVmConfig = New-AzVMConfig `  
    -VMName myVmWeb `  
    -VMSize Standard_DS1_V2 | `  
Set-AzVMOperatingSystem -Windows `  
    -ComputerName myVmWeb `  
    -Credential $cred | `  
Set-AzVMSourceImage `  
    -PublisherName MicrosoftWindowsServer `  
    -Offer WindowsServer `  
    -Skus 2016-Datacenter `  
    -Version latest | `  
Add-AzVMNetworkInterface `  
    -Id $webNic.Id  
New-AzVM `  
    -ResourceGroupName myResourceGroup `  
    -Location eastus `  
    -VM $webVmConfig `  
    -AsJob
```

Create a VM to serve as a management server:

```

# Create user object
$cred = Get-Credential -Message "Enter a username and password for the virtual machine."

# Create the web server virtual machine configuration and virtual machine.
$mgmtVmConfig = New-AzVMConfig `

    -VMName myVmMgmt `

    -VMSize Standard_DS1_V2 | `

Set-AzVMOperatingSystem -Windows `

    -ComputerName myVmMgmt `

    -Credential $cred | `

Set-AzVMSourceImage `

    -PublisherName MicrosoftWindowsServer `

    -Offer WindowsServer `

    -Skus 2016-Datacenter `

    -Version latest | `

Add-AzVMNetworkInterface `

    -Id $mgmtNic.Id

New-AzVM `

    -ResourceGroupName myResourceGroup `

    -Location eastus `

    -VM $mgmtVmConfig

```

The virtual machine takes a few minutes to create. Don't continue with the next step until Azure finishes creating the VM.

## Test traffic filters

Use [Get-AzPublicIpAddress](#) to return the public IP address of a VM. The following example returns the public IP address of the *myVmMgmt* VM:

```

Get-AzPublicIpAddress `

    -Name myVmMgmt `

    -ResourceGroupName myResourceGroup `

    | Select IpAddress

```

Use the following command to create a remote desktop session with the *myVmMgmt* VM from your local computer. Replace `<publicIpAddress>` with the IP address returned from the previous command.

```
mstsc /v:<publicIpAddress>
```

Open the downloaded RDP file. If prompted, select **Connect**.

Enter the user name and password you specified when creating the VM (you may need to select **More choices**, then **Use a different account**, to specify the credentials you entered when you created the VM), then select **OK**. You may receive a certificate warning during the sign-in process. Select **Yes** to proceed with the connection.

The connection succeeds, because port 3389 is allowed inbound from the internet to the *myAsgMgmtServers* application security group that the network interface attached to the *myVmMgmt* VM is in.

Use the following command to create a remote desktop connection to the *myVmWeb* VM, from the *myVmMgmt* VM, with the following command, from PowerShell:

```
mstsc /v:myvmWeb
```

The connection succeeds because a default security rule within each network security group allows traffic over all ports between all IP addresses within a virtual network. You can't create a remote desktop connection to the *myVmWeb* VM from the internet because the security rule for the *myAsgWebServers* doesn't allow port 3389

inbound from the internet.

Use the following command to install Microsoft IIS on the *myVmWeb* VM from PowerShell:

```
Install-WindowsFeature -name Web-Server -IncludeManagementTools
```

After the IIS installation is complete, disconnect from the *myVmWeb* VM, which leaves you in the *myVmMgmt* VM remote desktop connection. To view the IIS welcome screen, open an internet browser and browse to <http://myVmWeb>.

Disconnect from the *myVmMgmt* VM.

On your computer, enter the following command from PowerShell to retrieve the public IP address of the *myVmWeb* server:

```
Get-AzPublicIpAddress  
-Name myVmWeb  
-ResourceGroupName myResourceGroup  
| Select IpAddress
```

To confirm that you can access the *myVmWeb* web server from outside of Azure, open an internet browser on your computer and browse to <http://<public-ip-address-from-previous-step>>. The connection succeeds, because port 80 is allowed inbound from the internet to the *myAsgWebServers* application security group that the network interface attached to the *myVmWeb* VM is in.

## Clean up resources

When no longer needed, you can use [Remove-AzResourceGroup](#) to remove the resource group and all of the resources it contains:

```
Remove-AzResourceGroup -Name myResourceGroup -Force
```

## Next steps

In this article, you created a network security group and associated it to a virtual network subnet. To learn more about network security groups, see [Network security group overview](#) and [Manage a network security group](#).

Azure routes traffic between subnets by default. You may instead, choose to route traffic between subnets through a VM, serving as a firewall, for example. To learn how, see [Create a route table](#).

# Filter network traffic with a network security group using the Azure CLI

9/21/2022 • 7 minutes to read • [Edit Online](#)

You can filter network traffic inbound to and outbound from a virtual network subnet with a network security group. Network security groups contain security rules that filter network traffic by IP address, port, and protocol. Security rules are applied to resources deployed in a subnet. In this article, you learn how to:

- Create a network security group and security rules
- Create a virtual network and associate a network security group to a subnet
- Deploy virtual machines (VM) into a subnet
- Test traffic filters

If you don't have an [Azure subscription](#), create an [Azure free account](#) before you begin.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This article requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a network security group

A network security group contains security rules. Security rules specify a source and destination. Sources and destinations can be application security groups.

### Create application security groups

First create a resource group for all the resources created in this article with `az group create`. The following example creates a resource group in the `eastus` location:

```
az group create \
--name myResourceGroup \
--location eastus
```

Create an application security group with [az network asg create](#). An application security group enables you to group servers with similar port filtering requirements. The following example creates two application security groups.

```
az network asg create \
--resource-group myResourceGroup \
--name myAsgWebServers \
--location eastus

az network asg create \
--resource-group myResourceGroup \
--name myAsgMgmtServers \
--location eastus
```

## Create a network security group

Create a network security group with [az network nsg create](#). The following example creates a network security group named *myNsg*.

```
# Create a network security group
az network nsg create \
--resource-group myResourceGroup \
--name myNsg
```

## Create security rules

Create a security rule with [az network nsg rule create](#). The following example creates a rule that allows traffic inbound from the internet to the *myWebServers* application security group over ports 80 and 443:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNsg \
--name Allow-Web-All \
--access Allow \
--protocol Tcp \
--direction Inbound \
--priority 100 \
--source-address-prefix Internet \
--source-port-range "*" \
--destination-asgs "myAsgWebServers" \
--destination-port-range 80 443
```

The following example creates a rule that allows traffic inbound from the Internet to the *myMgmtServers* application security group over port 22:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNsg \
--name Allow-SSH-All \
--access Allow \
--protocol Tcp \
--direction Inbound \
--priority 110 \
--source-address-prefix Internet \
--source-port-range "*" \
--destination-asgs "myAsgMgmtServers" \
--destination-port-range 22
```

In this article, SSH (port 22) is exposed to the internet for the *myAsgMgmtServers* VM. For production environments, instead of exposing port 22 to the internet, it's recommended that you connect to Azure resources that you want to manage using a [VPN](#) or [private](#) network connection.

## Create a virtual network

Create a virtual network with [az network vnet create](#). The following example creates a virtual named *myVirtualNetwork*:

```
az network vnet create \
--name myVirtualNetwork \
--resource-group myResourceGroup \
--address-prefixes 10.0.0.0/16
```

Add a subnet to a virtual network with [az network vnet subnet create](#). The following example adds a subnet named *mySubnet* to the virtual network and associates the *myNsg* network security group to it:

```
az network vnet subnet create \
--vnet-name myVirtualNetwork \
--resource-group myResourceGroup \
--name mySubnet \
--address-prefix 10.0.0.0/24 \
--network-security-group myNsg
```

## Create virtual machines

Create two VMs in the virtual network so you can validate traffic filtering in a later step.

Create a VM with [az vm create](#). The following example creates a VM that will serve as a web server. The `--asgs myAsgWebServers` option causes Azure to make the network interface it creates for the VM a member of the *myAsgWebServers* application security group.

The `--nsg ""` option is specified to prevent Azure from creating a default network security group for the network interface Azure creates when it creates the VM. To streamline this article, a password is used. Keys are typically used in production deployments. If you use keys, you must also configure SSH agent forwarding for the remaining steps. For more information, see the documentation for your SSH client. Replace `<replace-with-your-password>` in the following command with a password of your choosing.

```
adminPassword=<replace-with-your-password>

az vm create \
--resource-group myResourceGroup \
--name myVmWeb \
--image UbuntuLTS \
--vnet-name myVirtualNetwork \
--subnet mySubnet \
--nsg "" \
--asgs myAsgWebServers \
--admin-username azureuser \
--admin-password $adminPassword
```

The VM takes a few minutes to create. After the VM is created, output similar to the following example is returned:

```
{
  "fqdns": "",
  "id": "/subscriptions/00000000-0000-0000-0000-
000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVmWeb",
  "location": "eastus",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
  "publicIpAddress": "13.90.242.231",
  "resourceGroup": "myResourceGroup"
}
```

Take note of the **publicIpAddress**. This address is used to access the VM from the internet in a later step. Create a VM to serve as a management server:

```
az vm create \
--resource-group myResourceGroup \
--name myVmMgmt \
--image UbuntuLTS \
--vnet-name myVirtualNetwork \
--subnet mySubnet \
--nsg "" \
--asgs myAsgMgmtServers \
--admin-username azureuser \
--admin-password $adminPassword
```

The VM takes a few minutes to create. After the VM is created, note the **publicIpAddress** in the returned output. This address is used to access the VM in the next step. Don't continue with the next step until Azure finishes creating the VM.

## Test traffic filters

Use the command that follows to create an SSH session with the *myVmMgmt* VM. Replace **<publicIpAddress>** with the public IP address of your VM. In the example above, the IP address is *13.90.242.231*.

```
ssh azureuser@<publicIpAddress>
```

When prompted for a password, enter the password you entered in [Create VMs](#).

The connection succeeds, because port 22 is allowed inbound from the Internet to the *myAsgMgmtServers* application security group that the network interface attached to the *myVmMgmt* VM is in.

Use the following command to SSH to the *myVmWeb* VM from the *myVmMgmt* VM:

```
ssh azureuser@myVmWeb
```

The connection succeeds because a default security rule within each network security group allows traffic over all ports between all IP addresses within a virtual network. You can't SSH to the *myVmWeb* VM from the Internet because the security rule for the *myAsgWebServers* doesn't allow port 22 inbound from the Internet.

Use the following commands to install the nginx web server on the *myVmWeb* VM:

```
# Update package source  
sudo apt-get -y update  
  
# Install NGINX  
sudo apt-get -y install nginx
```

The *myVmWeb* VM is allowed outbound to the Internet to retrieve nginx because a default security rule allows all outbound traffic to the Internet. Exit the *myVmWeb* SSH session, which leaves you at the `username@myVmMgmt:~$` prompt of the *myVmMgmt* VM. To retrieve the nginx welcome screen from the *myVmWeb* VM, enter the following command:

```
curl myVmWeb
```

Logout of the *myVmMgmt* VM. To confirm that you can access the *myVmWeb* web server from outside of Azure, enter `curl <publicIpAddress>` from your own computer. The connection succeeds, because port 80 is allowed inbound from the Internet to the *myAsgWebServers* application security group that the network interface attached to the *myVmWeb* VM is in.

## Clean up resources

When no longer needed, use [az group delete](#) to remove the resource group and all of the resources it contains.

```
az group delete --name myResourceGroup --yes
```

## Next steps

In this article, you created a network security group and associated it to a virtual network subnet. To learn more about network security groups, see [Network security group overview](#) and [Manage a network security group](#).

Azure routes traffic between subnets by default. You may instead, choose to route traffic between subnets through a VM, serving as a firewall, for example. To learn how, see [Create a route table](#).

# Create a virtual machine with a static public IP address using the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

A public IP address enables you to communicate to a virtual machine from the internet.

Assign a static public IP address, rather than a dynamic address, to ensure that the address never changes.

## Sign in to Azure

Sign in to the [Azure portal](#).

## Create a virtual machine

1. On the upper-left side of the portal, select **Create a resource > Compute > Virtual machine** or search for **Virtual machine** in the search box.
2. In **Create a virtual machine**, type or select the values in the **Basics** tab:

SETTING	VALUE
Project Details	
Subscription	Select your Azure subscription
Resource Group	Select <b>Create new</b> . In <b>Name</b> , enter <b>myResourceGroup</b> . Select <b>OK</b> .
Instance details	
Virtual machine name	Enter <b>myVM</b>
Region	Select <b>East US</b>
Availability Options	Select <b>No infrastructure redundancy required</b>
Image	Select <b>Windows Server 2019 Datacenter - Gen1</b>
Azure Spot instance	Select <b>No</b>
Size	Choose VM size or take default setting
Administrator account	
Username	Enter a username
Password	Enter a password

SETTING	VALUE
Confirm password	Reenter password
Public inbound ports	Select <b>Allow selected ports</b> .
Select inbound ports	Select RDP (3389)

**WARNING**

Portal 3389 is selected, to enable remote access to the Windows Server virtual machine from the internet.

Opening port 3389 to the internet is not recommended to manage production workloads.

For secure access to Azure virtual machines, see [What is Azure Bastion?](#)

3. Select the **Networking** tab, or select **Next: Disks**, then **Next: Networking**.

4. In the Networking tab, select or enter:

SETTING	VALUE
<b>Network interface</b>	
Virtual network	Accept the default network name.
Subnet	Accept the default subnet configuration.
Public IP	Select <b>Create new</b> . In <b>Create public IP address</b> , in name enter <b>myPublicIP</b> . For <b>SKU</b> , select <b>Standard</b> . <b>Assignment</b> , select <b>Static</b> . Select <b>OK</b> .
NIC network security group	Select <b>Basic</b>
Public inbound ports	Select <b>Allow selected ports</b> .
Select inbound ports	Select RDP (3389)

5. Select **Review + create**.

6. Review the settings, and then select **Create**.

#### **NOTE**

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Clean up resources

When no longer needed, delete the resource group and all of the resources it contains:

1. Enter **myResourceGroup** in the **Search** box at the top of the portal. When you see **myResourceGroup** in the search results, select it.
2. Select **Delete resource group**.
3. Enter **myResourceGroup** for **TYPE THE RESOURCE GROUP NAME:** and select **Delete**.

## Next steps

See [Add, change, or remove IP addresses](#):

- To change a public IP address from dynamic to static.
- Work with private IP addresses.

Public IP addresses have a [nominal charge](#). There's a [limit](#) to the number of public IP addresses that you can use per subscription.

The SKU of the virtual machine's public IP address must match the public IP SKU of Azure Load Balancer when added to a backend pool. For details, see [Azure Load Balancer](#).

You can download the list of ranges (prefixes) for the Azure [Public](#), [US government](#), [China](#), and [Germany](#) clouds.

- Learn more about [static public IP addresses](#).
- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.

# Create a virtual machine with a static public IP address using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

In this article, you'll create a VM with a static public IP address. A public IP address enables communication to a virtual machine from the internet. Assign a static public IP address, instead of a dynamic address, to ensure the address never changes.

Public IP addresses have a [nominal charge](#). There's a [limit](#) to the number of public IP addresses that you can use per subscription.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- Azure PowerShell installed locally or Azure Cloud Shell

If you choose to install and use PowerShell locally, this article requires the Azure PowerShell module version 5.4.1 or later. Run `Get-Module -ListAvailable Az` to find the installed version. If you need to upgrade, see [Install Azure PowerShell module](#). If you're running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [New-AzResourceGroup](#) named **myResourceGroup** in the **eastus2** location.

```
$rg =@{  
    Name = 'myResourceGroup'  
    Location = 'eastus2'  
}  
New-AzResourceGroup @rg
```

## Create a public IP address

Use [New-AzPublicIpAddress](#) to create a standard public IPv4 address.

The following command creates a zone-redundant public IP address named **myPublicIP** in **myResourceGroup**.

```

## Create IP. ##
$ip = @{
    Name = 'myPublicIP'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    Sku = 'Standard'
    AllocationMethod = 'Static'
   IpAddressVersion = 'IPv4'
    Zone = 1,2,3
}
New-AzPublicIpAddress @ip

```

## Create a virtual machine

Create a virtual machine with [New-AzVM](#).

The following command creates a Windows Server virtual machine. You'll enter the name of the public IP address created previously in the `-PublicIpAddressName` parameter. When prompted, provide a username and password to be used as the credentials for the virtual machine:

```

## Create virtual machine. ##
$vm = @{
    ResourceGroupName = 'myResourceGroup'
    Location = 'East US 2'
    Name = 'myVM'
    PublicIpAddressName = 'myPublicIP'
}
New-AzVM @vm

```

For more information on public IP SKUs, see [Public IP address SKUs](#). A virtual machine can be added to the backend pool of an Azure Load Balancer. The SKU of the public IP address must match the SKU of a load balancer's public IP. For more information, see [Azure Load Balancer](#).

View the public IP address assigned and confirm that it was created as a static address, with [Get-AzPublicIpAddress](#):

```

## Retrieve public IP address settings. ##
$ip = @{
    Name = 'myPublicIP'
    ResourceGroupName = 'myResourceGroup'
}
Get-AzPublicIpAddress @ip | Select "IpAddress","PublicIpAllocationMethod" | Format-Table

```

### WARNING

Do not modify the IP address settings within the virtual machine's operating system. The operating system is unaware of Azure public IP addresses. Though you can add private IP address settings to the operating system, we recommend not doing so unless necessary, and not until after reading [Add a private IP address to an operating system](#).

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Clean up resources

When no longer needed, you can use [Remove-AzResourceGroup](#) to remove the resource group and all of the resources it contains:

```
Remove-AzResourceGroup -Name myResourceGroup -Force
```

## Next steps

- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.
- Learn more about creating [Linux](#) and [Windows](#) virtual machines.

# Create a virtual machine with a static public IP address using the Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

In this article, you'll create a VM with a static public IP address. A public IP address enables communication to a virtual machine from the internet. Assign a static public IP address, instead of a dynamic address, to ensure the address never changes.

Public IP addresses have a [nominal charge](#). There's a [limit](#) to the number of public IP addresses that you can use per subscription.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).
- An Azure account with an active subscription. [Create an account for free](#).
- This tutorial requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [az group create](#) named **myResourceGroup** in the **eastus2** location.

```
az group create \
--name myResourceGroup \
--location eastus2
```

## Create a public IP address

Use [az network public-ip create](#) to create a standard public IPv4 address.

The following command creates a zone-redundant public IP address named **myPublicIP** in

myResourceGroup.

```
az network public-ip create \
--resource-group myResourceGroup \
--name myPublicIP \
--version IPv4 \
--sku Standard \
--zone 1 2 3
```

## Create a virtual machine

Create a virtual machine with [az vm create](#).

The following command creates a Windows Server virtual machine. You'll enter the name of the public IP address created previously in the `-PublicIPAddressName` parameter. When prompted, provide a username and password to be used as the credentials for the virtual machine:

```
az vm create \
--name myVM \
--resource-group TutorVMRoutePref-rg \
--public-ip-address myPublicIP \
--size Standard_A2 \
--image MicrosoftWindowsServer:WindowsServer:2019-Datacenter:latest \
--admin-username azureuser
```

For more information on public IP SKUs, see [Public IP address SKUs](#). A virtual machine can be added to the backend pool of an Azure Load Balancer. The SKU of the public IP address must match the SKU of a load balancer's public IP. For more information, see [Azure Load Balancer](#).

View the public IP address assigned and confirm that it was created as a static address, with [az network public-ip show](#):

```
az network public-ip show \
--resource-group myResourceGroup \
--name myPublicIP \
--query [ipAddress,publicIpAllocationMethod,sku] \
--output table
```

### WARNING

Do not modify the IP address settings within the virtual machine's operating system. The operating system is unaware of Azure public IP addresses. Though you can add private IP address settings to the operating system, we recommend not doing so unless necessary, and not until after reading [Add a private IP address to an operating system](#).

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Clean up resources

When no longer needed, you can use `az group delete` to remove the resource group and all of the resources it contains:

```
az group delete --name myResourceGroup --yes
```

## Next steps

- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.
- Learn more about creating [Linux](#) and [Windows](#) virtual machines.

# Associate a public IP address to a virtual machine

9/21/2022 • 11 minutes to read • [Edit Online](#)

In this article, you learn how to associate a public IP address to an existing virtual machine (VM). If you want to connect to a VM from the internet, the VM must have a public IP address associated to it. If you want to create a new VM with a public IP address, you can do so using the [Azure portal](#), the [Azure CLI](#), or [Azure PowerShell](#). Public IP addresses have a nominal fee. For details, see [pricing](#). There is a limit to the number of public IP addresses that you can use per subscription. For details, see [limits](#).

You can use the [Azure portal](#), the [Azure CLI](#), or [Azure PowerShell](#) to associate a public IP address to a VM.

## NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

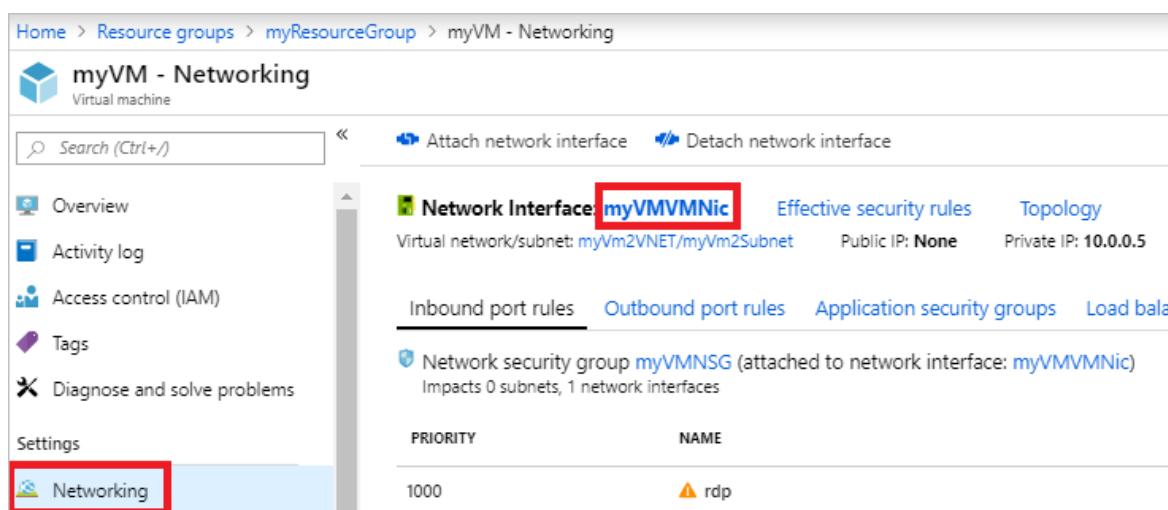
The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Azure portal

1. Sign in to the [Azure portal](#).
2. Browse to, or search for the virtual machine that you want to add the public IP address to and then select it.
3. Under **Settings**, select **Networking**, and then select the network interface you want to add the public IP address to, as shown in the following picture:



The screenshot shows the Azure portal interface for managing a virtual machine named 'myVM'. The 'Networking' section is open, displaying the 'myVMVMNic' network interface. Key details shown include:

- Virtual network/subnet:** myVm2VNET/myVm2Subnet
- Public IP:** None
- Private IP:** 10.0.0.5

The 'Inbound port rules' tab is selected, showing one rule:

PRIORITY	NAME
1000	rdp

Other tabs visible include 'Effective security rules' and 'Topology'. The left sidebar shows other settings like 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', and 'Networking' (which is highlighted with a red box).

#### NOTE

Public IP addresses are associated to network interfaces attached to a VM. In the previous picture, the VM only has one network interface. If the VM had multiple network interfaces, they would all appear, and you'd select the network interface you want to associate the public IP address to.

4. Select **IP configurations** and then select an IP configuration, as shown in the following picture:

Home > myVMVMNic - IP configurations

## myVMVMNic - IP configurations

Network interface

Search (Ctrl+ /) Add Save Discard

Overview Activity log Access control (IAM) Tags

Settings

IP configurations

Subnet: myVm2Subnet (10.0.0.0/24)

NAME	IP VERSIO...	TYPE	PRIVATE IP ADDRESS	PUBLIC IP ADDRESS
ipconfigmyVM	IPv4	Primary	10.0.0.5 (Dynamic)	-

#### NOTE

Public IP addresses are associated to IP configurations for a network interface. In the previous picture, the network interface has one IP configuration. If the network interface had multiple IP configurations, they would all appear in the list, and you'd select the IP configuration that you want to associate the public IP address to.

5. Select **Enabled**, then select **IP address (Configure required settings)**. Choose an existing public IP address, which automatically closes the **Choose public IP address** box. If you don't have any available public IP addresses listed, you need to create one. To learn how, see [Create a public IP address](#). Select **Save**, as shown in the picture that follows, and then close the box for the IP configuration.

Home > myVMVMNic - IP configurations > ipconfigmyVM > Choose public IP address

### ipconfigmyVM

Save Discard

Public IP address settings

Enabled

IP address Configure required settings

Choose public IP address

These are the public IP addresses in the selected subscription and location 'East US'.

Create new

myVMPublicIP myResourceGroup

#### NOTE

The public IP addresses that appear are those that exist in the same region as the VM. If you have multiple public IP addresses created in the region, all will appear here. If any are grayed out, it's because the address is already associated to a different resource.

6. View the public IP address assigned to the IP configuration, as shown in the picture that follows. It may take a few seconds for an IP address to appear.

The screenshot shows the Azure portal's 'IP configurations' page for a network interface named 'myVMVMNic'. On the left, there's a sidebar with options like Overview, Activity log, Access control (IAM), Tags, Settings, IP configurations (which is selected), DNS servers, Network security group, and Properties. The main area shows 'IP forwarding settings' with 'IP forwarding' set to 'Enabled'. Below that is a section for 'Virtual network' and 'myVm2VNET'. Under 'IP configurations', there's a dropdown for 'Subnet' set to 'myVm2Subnet (10.0.0.0/24)'. A table lists one IP configuration: 'ipconfig...' (IPv4), Type 'Primary', Private IP Address '10.0.0.5 (Dynamic)', and Public IP Address '52.179.3.114 (myVMPublicIP)'. The public IP address is highlighted with a red box.

#### NOTE

The address is assigned from a pool of addresses used in each Azure region. To see a list of address pools used in each region, see [Microsoft Azure Datacenter IP Ranges](#). The address assigned can be any address in the pools used for the region. If you need the address to be assigned from a specific pool in the region, use a [Public IP address prefix](#).

7. [Allow network traffic to the VM](#) with security rules in a network security group.

## Azure CLI

Install the [Azure CLI](#), or use the Azure Cloud Shell. The Azure Cloud Shell is a free Bash shell that you can run directly within the Azure portal. It has the Azure CLI preinstalled and configured to use with your account. Select the Try it button in the CLI commands that follow. Selecting Try it invokes a Cloud Shell that you can sign in to your Azure account with.

1. If using the CLI locally in Bash, sign in to Azure with `az login`.
2. A public IP address is associated to an IP configuration of a network interface attached to a VM. Use the `az network nic-ip-config update` command to associate a public IP address to an IP configuration. The following example associates an existing public IP address named *myVMPublicIP* to the IP configuration named *ipconfigmyVM* of an existing network interface named *myVMVMNic* that exists in a resource group named *myResourceGroup*.

```
az network nic ip-config update \
--name ipconfigmyVM \
--nic-name myVMVMNic \
--resource-group myResourceGroup \
--public-ip-address myVMPublicIP
```

- If you don't have an existing public IP address, use the `az network public-ip create` command to create one. For example, the following command creates a public IP address named *myVMPublicIP* in a resource group named *myResourceGroup*.

```
az network public-ip create --name myVMPublicIP --resource-group myResourceGroup
```

#### NOTE

The previous command creates a public IP address with default values for several settings that you may want to customize. To learn more about all public IP address settings, see [Create a public IP address](#). The address is assigned from a pool of public IP addresses used for each Azure region. To see a list of address pools used in each region, see [Microsoft Azure Datacenter IP Ranges](#).

- If you don't know the name of a network interface attached to your VM, use the [az vm nic list](#) command to view them. For example, the following command lists the names of the network interfaces attached to a VM named *myVM* in a resource group named *myResourceGroup*:

```
az vm nic list --vm-name myVM --resource-group myResourceGroup
```

The output includes one or more lines that are similar to the following example:

```
"id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVMVMNic",
```

In the previous example, *myVMVMNic* is the name of the network interface.

- If you don't know the name of an IP configuration for a network interface, use the [az network nic ip-config list](#) command to retrieve them. For example, the following command lists the names of the IP configurations for a network interface named *myVMVMNic* in a resource group named *myResourceGroup*:

```
az network nic ip-config list --nic-name myVMVMNic --resource-group myResourceGroup --out table
```

3. View the public IP address assigned to the IP configuration with the [az vm list-ip-addresses](#) command. The following example shows the IP addresses assigned to an existing VM named *myVM* in a resource group named *myResourceGroup*.

```
az vm list-ip-addresses --name myVM --resource-group myResourceGroup --out table
```

#### NOTE

The address is assigned from a pool of addresses used in each Azure region. To see a list of address pools used in each region, see [Microsoft Azure Datacenter IP Ranges](#). The address assigned can be any address in the pools used for the region. If you need the address to be assigned from a specific pool in the region, use a [Public IP address prefix](#).

4. [Allow network traffic to the VM](#) with security rules in a network security group.

## PowerShell

Install [PowerShell](#), or use the Azure Cloud Shell. The Azure Cloud Shell is a free shell that you can run directly within the Azure portal. It has PowerShell preinstalled and configured to use with your account. Select the **Try it** button in the PowerShell commands that follow. Selecting **Try it** invokes a Cloud Shell that you can sign in to your Azure account with.

1. If using PowerShell locally, sign in to Azure with `Connect-AzAccount`.
2. A public IP address is associated to an IP configuration of a network interface attached to a VM. Use the [Get-AzVirtualNetwork](#) and [Get-AzVirtualNetworkSubnetConfig](#) commands to get the virtual network and subnet that the network interface is in. Next, use the [Get-AzNetworkInterface](#) command to get a network interface and the [Get-AzPublicIpAddress](#) command to get an existing public IP address. Then use the [Set-AzNetworkInterfaceIpConfig](#) command to associate the public IP address to the IP configuration and the [Set-AzNetworkInterface](#) command to write the new IP configuration to the network interface.

The following example associates an existing public IP address named *myVMPublicIP* to the IP configuration named *ipconfigmyVM* of an existing network interface named *myVMVNic* that exists in a subnet named *myVMSubnet* in a virtual network named *myVMVNet*. All resources are in a resource group named *myResourceGroup*.

```
$vnet = Get-AzVirtualNetwork -Name myVMVNet -ResourceGroupName myResourceGroup
$subnet = Get-AzVirtualNetworkSubnetConfig -Name myVMSubnet -VirtualNetwork $vnet
$nic = Get-AzNetworkInterface -Name myVMVNic -ResourceGroupName myResourceGroup
$pip = Get-AzPublicIpAddress -Name myVMPublicIP -ResourceGroupName myResourceGroup
$nic | Set-AzNetworkInterfaceIpConfig -Name ipconfigmyVM -PublicIPAddress $pip -Subnet $subnet
$nic | Set-AzNetworkInterface
```

- If you don't have an existing public IP address, use the [New-AzPublicIpAddress](#) command to create one. For example, the following command creates a *dynamic* public IP address named *myVMPublicIP* in a resource group named *myResourceGroup* in the *eastus* region.

```
New-AzPublicIpAddress -Name myVMPublicIP -ResourceGroupName myResourceGroup -AllocationMethod Dynamic -Location eastus
```

#### NOTE

The previous command creates a public IP address with default values for several settings that you may want to customize. To learn more about all public IP address settings, see [Create a public IP address](#). The address is assigned from a pool of public IP addresses used for each Azure region. To see a list of address pools used in each region, see [Microsoft Azure Datacenter IP Ranges](#).

- If you don't know the name of a network interface attached to your VM, use the [Get-AzVM](#) command to view them. For example, the following command lists the names of the network interfaces attached to a VM named *myVM* in a resource group named *myResourceGroup*.

```
$vm = Get-AzVM -name myVM -ResourceGroupName myResourceGroup
$vm.NetworkProfile
```

The output includes one or more lines that are similar to the example that follows. In the example output, *myVMVNic* is the name of the network interface.

```
"id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVMVNic",
```

- If you don't know the name of the virtual network or subnet that the network interface is in, use the [Get-AzNetworkInterface](#) command to view the information. For example, the following command gets the virtual network and subnet information for a network interface named *myVMVNic* in a resource group named *myResourceGroup*.

```
$nic = Get-AzNetworkInterface -Name myVMVMNic -ResourceGroupName myResourceGroup  
$ipConfigs = $nic.IpConfigurations  
$ipConfigs.Subnet | Select Id
```

The output includes one or more lines that are similar to the example that follows. In the example output, *myVMVNET* is the name of the virtual network and *myVMSubnet* is the name of the subnet.

```
"/subscriptions/00000000-0000-0000-0000-  
0000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/virtualNetworks/myVMV-  
NET/subnets/myVMSubnet",
```

- If you don't know the name of an IP configuration for a network interface, use the [Get-AzNetworkInterface](#) command to retrieve them. For example, the following command lists the names of the IP configurations for a network interface named *myVMVMNic* in a resource group named *myResourceGroup*.

```
$nic = Get-AzNetworkInterface -Name myVMVMNic -ResourceGroupName myResourceGroup  
$nic.IPCConfigurations
```

The output includes one or more lines that are similar to the example that follows. In the example output, *ipconfigmyVM* is the name of an IP configuration.

```
Id      : /subscriptions/00000000-0000-0000-0000-  
0000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVM-  
VMNic/ipConfigurations/ipconfigmyVM
```

3. View the public IP address assigned to the IP configuration with the [Get-AzPublicIpAddress](#) command. The following example shows the address assigned to a public IP address named *myVMPublicIP* in a resource group named *myResourceGroup*.

```
Get-AzPublicIpAddress -Name myVMPublicIP -ResourceGroupName myResourceGroup | Select IpAddress
```

If you don't know the name of the public IP address assigned to an IP configuration, run the following commands to get it:

```
$nic = Get-AzNetworkInterface -Name myVMVMNic -ResourceGroupName myResourceGroup  
$nic.IPCConfigurations  
$address = $nic.IPCConfigurations.PublicIpAddress  
$address | Select Id
```

The output includes one or more lines that are similar to the example that follows. In the example output, *myVMPublicIP* is the name of the public IP address assigned to the IP configuration.

```
"/subscriptions/00000000-0000-0000-0000-  
0000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/publicIPAddresses/myVMPublicI-  
P"
```

**NOTE**

The address is assigned from a pool of addresses used in each Azure region. To see a list of address pools used in each region, see [Microsoft Azure Datacenter IP Ranges](#). The address assigned can be any address in the pools used for the region. If you need the address to be assigned from a specific pool in the region, use a [Public IP address prefix](#).

4. [Allow network traffic to the VM](#) with security rules in a network security group.

## Allow network traffic to the VM

Before you can connect to the public IP address from the internet, ensure that you have the necessary ports open in any network security group that you might have associated to the network interface, the subnet the network interface is in, or both. Though security groups filter traffic to the private IP address of the network interface, once inbound internet traffic arrives at the public IP address, Azure translates the public address to the private IP address, so if a network security group prevents the traffic flow, the communication with the public IP address fails. You can view the effective security rules for a network interface and its subnet using the [Portal](#), [CLI](#), or [PowerShell](#).

## Next steps

Allow inbound internet traffic to your VM with a network security group. To learn how to create a network security group, see [Work with network security groups](#). To learn more about network security groups, see [Security groups](#).

# Dissociate a public IP address from an Azure VM

9/21/2022 • 4 minutes to read • [Edit Online](#)

In this article, you learn how to dissociate a public IP address from an Azure virtual machine (VM).

You can use the [Azure portal](#), the [Azure CLI](#), or [Azure PowerShell](#) to dissociate a public IP address from a VM.

## Azure portal

1. Sign in to the [Azure portal](#).
2. Browse to, or search for the virtual machine that you want to disassociate the public IP address from and then select it.
3. In the VM page, select **Overview**, select the public IP address as shown in the following picture:

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. Below the navigation is a breadcrumb trail: 'Dashboard > Resource groups > myResourceGroup > myVM'. The main content area has a left sidebar with options like 'Search (Ctrl+J)', 'Connect', 'Start', 'Restart', 'Stop', 'Capture', 'Delete', 'Refresh', and a link to 'Advisor (1 of 3): Enable virtual machine replication to protect your applications from regional outage'. The main panel is titled 'myVM Virtual machine' and shows the 'Overview' tab selected. It displays details such as Resource group (changed), Status, Location, Subscription, Computer name, Operating system, Size, Tags, and several IP addresses (Public and Private). The 'Public IP address' field is explicitly highlighted with a red box, containing the value '52.170.252.185'.

4. In the public IP address page, select **Overview**, and then select **Dissociate**, as shown in the following picture:

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes 'Microsoft Azure', a search bar, and various icons. Below the navigation is a breadcrumb trail: 'Dashboard > Resource groups > myResourceGroup > myVM > myPublicIpAddress'. The main content area has a left sidebar with options like 'Search (Ctrl+J)', 'Associate', 'Dissociate' (which is highlighted with a red box), 'Move', 'Delete', and 'Refresh'. The main panel is titled 'myPublicIpAddress Public IP address' and shows the 'Overview' tab selected. It displays details such as Resource group, Location, Subscription, SKU, IP address, DNS name, Associated to, and Virtual machine.

5. In **Dissociate public IP address**, select **Yes**.

## Azure CLI

Install the [Azure CLI](#), or use the Azure Cloud Shell. The Azure Cloud Shell is a free Bash shell that you can run directly within the Azure portal. It has the Azure CLI preinstalled and configured to use with your account. Select the **Try it** button in the CLI commands that follow. Selecting **Try it** invokes a Cloud Shell that you can sign in to your Azure account with.

1. If using the CLI locally in Bash, sign in to Azure with `az login`.
2. A public IP address is associated to an IP configuration of a network interface attached to a VM. Use the `az network nic-ip-config update` command to dissociate a public IP address from an IP configuration. The

following example dissociates a public IP address named *myVMPublicIP* from the IP configuration named *ipconfigmyVM* of an existing network interface named *myVMVMNic* that is attached to a VM named *myVM* in a resource group named *myResourceGroup*.

```
az network nic ip-config update \
--name ipconfigmyVM \
--resource-group myResourceGroup \
--nic-name myVMVMNic \
--remove PublicIpAddress
```

If you don't know the name of a network interface attached to your VM, use the [az vm nic list](#) command to view them. For example, the following command lists the names of the network interfaces attached to a VM named *myVM* in a resource group named *myResourceGroup*.

```
az vm nic list --vm-name myVM --resource-group myResourceGroup
```

The output includes one or more lines that are similar to the following example:

```
"id": "/subscriptions/00000000-0000-0000-0000-000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVMVMNic",
```

In the previous example, *myVMVMNic* is the name of the network interface.

- If you don't know the name of an IP configuration for a network interface, use the [az network nic ip-config list](#) command to retrieve them. For example, the following command lists the names of the public IP configurations for a network interface named *myVMVMNic* in a resource group named *myResourceGroup*.

```
az network nic ip-config list --nic-name myVMVMNic --resource-group myResourceGroup --out table
```

- If you don't know the name of a public IP configuration for a network interface, use the [az network nic ip-config show](#) command to retrieve them. For example, the following command lists the names of the public IP configurations for a network interface named *myVMVMNic* in a resource group named *myResourceGroup*.

```
az network nic ip-config show --name ipconfigmyVM --nic-name myVMVMNic --resource-group myResourceGroup --query publicIPAddress.id
```

## PowerShell

Install [PowerShell](#), or use the Azure Cloud Shell. The Azure Cloud Shell is a free shell that you can run directly within the Azure portal. It has PowerShell preinstalled and configured to use with your account. Select the **Try it** button in the PowerShell commands that follow. Selecting **Try it** invokes a Cloud Shell that you can sign in to your Azure account with.

1. If using PowerShell locally, sign in to Azure with `Connect-AzAccount`.
2. A public IP address is associated to an IP configuration of a network interface attached to a VM. Use the [Get-AzNetworkInterface](#) command to get a network interface. Set the Public IP address value to null and then use the [Set-AzNetworkInterface](#) command to write the new IP configuration to the network interface.

The following example dissociates a public IP address named *myVMPublicIP* from a network interface named *myVMVMNic* that is attached to a VM named *myVM*. All resources are in a resource group named *myResourceGroup*.

```
$nic = Get-AzNetworkInterface -Name myVMVMNic -ResourceGroupName myResourceGroup  
$nic.IpConfigurations.publicipaddress.id = $null  
Set-AzNetworkInterface -NetworkInterface $nic
```

- If you don't know the name of a network interface attached to your VM, use the [Get-AzVM](#) command to view them. For example, the following command lists the names of the network interfaces attached to a VM named *myVM* in a resource group named *myResourceGroup*.

```
$vm = Get-AzVM -name myVM -ResourceGroupName myResourceGroup  
$vm.NetworkProfile
```

The output includes one or more lines that are similar to the example that follows. In the example output, *myVMVMNic* is the name of the network interface.

```
"id": "/subscriptions/00000000-0000-0000-0000-  
000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVMVMNic",
```

- If you don't know the name of an IP configuration for a network interface, use the [Get-AzNetworkInterface](#) command to retrieve them. For example, the following command lists the names of the IP configurations for a network interface named *myVMVMNic* in a resource group named *myResourceGroup*.

```
$nic = Get-AzNetworkInterface -Name myVMVMNic -ResourceGroupName myResourceGroup  
$nic.IPCConfigurations.id
```

The output includes one or more lines that are similar to the example that follows. In the example output, *ipconfigmyVM* is the name of an IP configuration.

```
"id": "/subscriptions/00000000-0000-0000-0000-  
000000000000/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myVMVMNic/i  
pConfigurations/ipconfigmyVM"
```

## Next steps

- Learn how to [associate a public IP address to a VM](#).

# Create a virtual machine with a static private IP address using the Azure portal

9/21/2022 • 4 minutes to read • [Edit Online](#)

A virtual machine (VM) is automatically assigned a private IP address from a range that you specify. This range is based on the subnet in which the VM is deployed. The VM keeps the address until the VM is deleted. Azure dynamically assigns the next available private IP address from the subnet you create a VM in. Assign a static IP address to the VM if you want a specific IP address in the subnet.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).

## Create virtual machine

Use the following steps to create a virtual machine, virtual network, and subnet.

1. Sign in to the [Azure portal](#).
2. In the search box at the top of the portal, enter **Virtual machine**. Select **Virtual machines** in the search results.
3. Select + **Create**, then + **Virtual machine** in **Virtual machines**.
4. In **Create a virtual machine**, enter or select the following information:

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>Create new</b> . Enter <b>myResourceGroup</b> in <b>Name</b> . Select <b>OK</b> .
<b>Instance details</b>	
Virtual machine name	Enter <b>myVM</b> .
Region	Select <b>(US) East US 2</b> .
Availability options	Select <b>No infrastructure redundancy required</b> .
Image	Select <b>Windows Server 2019 Datacenter - Gen2</b> .
Azure Spot instance	Leave unchecked.
Size	Select a size.

SETTING	VALUE
Administrator account	
Username	Enter a username.
Password	Enter a password.
Confirm password	Reenter password.
Public inbound ports	Select <b>Allow selected ports</b> .
Select inbound ports	Select RDP (3389)

The screenshot shows the Microsoft Azure portal interface for creating a new virtual machine. The left sidebar contains a navigation menu with various service icons. The main content area is titled 'Create a virtual machine' and is currently on the 'Basics' tab. The page includes sections for 'Project details', 'Instance details', 'Administrator account', and 'Inbound port rules'. A warning message at the bottom of the 'Inbound port rules' section states: '⚠️ This will allow all IP addresses to access your virtual machine. This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.' Navigation buttons at the bottom include 'Review + create', '< Previous', and 'Next : Disks >'.

### WARNING

Portal 3389 is selected, to enable remote access to the Windows Server virtual machine from the internet.  
Opening port 3389 to the internet is not recommended to manage production workloads.  
For secure access to Azure virtual machines, see [What is Azure Bastion?](#)

5. Select the **Networking** tab, or select **Next: Disks**, then **Next: Networking**.

6. In the **Networking** tab, select or enter:

SETTING	VALUE
<b>Network interface</b>	
Virtual network	Accept the default network name.
Subnet	Accept the default subnet configuration.
Public IP	Accept the default public IP configuration.
NIC network security group	Select <b>Basic</b>
Public inbound ports	Select <b>Allow selected ports</b> .
Select inbound ports	Select <b>RDP (3389)</b>

7. Select **Review + create**.

8. Review the settings, and then select **Create**.

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Change private IP address to static

In this section, you'll change the private IP address from **dynamic** to **static** for the virtual machine you created previously.

1. In the search box at the top of the portal, enter **Virtual machine**. Select **Virtual machines** in the search results.
2. In **Virtual machines**, select **myVM**.
3. Select **Networking** in **Settings** in **myVM**.
4. In **Networking**, select the name of the network interface next to **Network interface**. In this example, the name of the NIC is **myvm472**.

myVM | Networking

Virtual machine

Search (Ctrl+ /) < Attach network interface Detach network interface

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings

**Networking**

Connect Windows Admin Center (preview) Disks Size Security Advisor recommendations Extensions

myvm472

IP configuration ipconfig1 (Primary)

Network Interface: myvm472 Effective security rules Troubleshoot VM connection issues Topology

Virtual network/subnet: myResourceGroup-vnet/default NIC Public IP: 20.62.69.9 NIC Private IP: 10.1.0.4 Accelerated networking: Enabled

Inbound port rules Outbound port rules Application security groups Load balancing

Network security group myVM-nsg (attached to network interface: myvm472)  
Impacts 0 subnets, 1 network interfaces Add

Priority	Name	Port	Protocol	Source
300	RDP	3389	TCP	Any
65000	AllowVnetInBound	Any	Any	VirtualNetwork
65001	AllowAzureLoadBalancerInB...	Any	Any	AzureLoadBalanc...
65500	DenyAllInBound	Any	Any	Any

5. In the network interface properties, select **IP configurations** in **Settings**.

6. Select **ipconfig1** in the **IP configurations** page.

Home > Virtual machines > myVM > myvm472

myvm472 | IP configurations

Network interface

Search (Ctrl+ /) + Add Save Discard Refresh

Overview Activity log Access control (IAM) Tags

Settings

**IP configurations**

IP forwarding settings IP forwarding **Enabled**

Virtual network myResourceGroup-vnet

IP configurations Subnet \*

default (10.1.0.0/24)

Search IP configurations

Name	IP Version	Type	Private IP address	Public IP address	...
ipconfig1	IPv4	Primary	10.1.0.4 (Dynamic)	20.62.69.9 (myVM-ip)	...

7. Select **Static** in **Assignment**. Select **Save**.

ipconfig1 ... X

myvm472

Save Discard

Public IP address settings

Public IP address

Disassociate Associate

Public IP address \*

myVM-ip (20.62.69.9) ▼

[Create new](#)

Private IP address settings

Virtual network/subnet  
myResourceGroup-vnet/default

Assignment

Dynamic Static

IP address \*

10.1.0.4

**NOTE**

If you notice after selecting **Save** that the assignment is still set to **Dynamic**, the IP address you typed is already in use. Try another IP address.

To change the IP address back to dynamic set the assignment for your private IP address to **Dynamic**, and then select **Save**.

**WARNING**

From within the operating system of a VM, you shouldn't statically assign the *private IP* that's assigned to the Azure VM. Only do static assignment of a private IP when it's necessary, such as when [assigning many IP addresses to VMs](#).

If you manually set the private IP address within the operating system, make sure it matches the private IP address assigned to the Azure [network interface](#). Otherwise, you can lose connectivity to the VM. Learn more about [private IP address settings](#).

## Clean up resources

When no longer needed, delete the resource group and all of the resources it contains:

1. Enter **myResourceGroup** in the **Search** box at the top of the portal. When you see **myResourceGroup** in the search results, select it.
2. Select **Delete resource group**.
3. Enter **myResourceGroup** for **TYPE THE RESOURCE GROUP NAME:** and select **Delete**.

## Next steps

- Learn more about [static public IP addresses](#).
- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.

# Create a virtual machine with a static private IP address using Azure PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

A virtual machine (VM) is automatically assigned a private IP address from a range that you specify. This range is based on the subnet in which the VM is deployed. The VM keeps the address until the VM is deleted. Azure dynamically assigns the next available private IP address from the subnet you create a VM in. Assign a static IP address to the VM if you want a specific IP address in the subnet.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- Azure PowerShell installed locally or Azure Cloud Shell

If you choose to install and use PowerShell locally, this article requires the Azure PowerShell module version 5.4.1 or later. Run `Get-Module -ListAvailable Az` to find the installed version. If you need to upgrade, see [Install Azure PowerShell module](#). If you're running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [New-AzResourceGroup](#) named **myResourceGroup** in the **eastus2** location.

```
$rg =@{  
    Name = 'myResourceGroup'  
    Location = 'eastus2'  
}  
New-AzResourceGroup @rg
```

## Create a virtual machine

Create a virtual machine with [New-AzVM](#).

The following command creates a Windows Server virtual machine. When prompted, provide a username and password to be used as the credentials for the virtual machine:

```
## Create virtual machine. ##  
$vm = @{  
    ResourceGroupName = 'myResourceGroup'  
    Location = 'East US 2'  
    Name = 'myVM'  
    PublicIpAddressName = 'myPublicIP'  
}  
New-AzVM @vm
```

## Change private IP address to static

In this section, you'll change the private IP address from **dynamic** to **static** for the virtual machine you created previously.

Use [Get-AzVirtualNetwork](#) to place the virtual network configuration into a variable. Use [Get-AzVirtualNetworkSubnetConfig](#) to place the subnet configuration into a variable. Use [Get-AzNetworkInterface](#) to obtain the network interface configuration and place into a variable. Use [Set-AzNetworkInterfaceIpConfig](#) to set the configuration of the network interface. Finally, use [Set-AzNetworkInterface](#) to set the configuration for the virtual machine.

The following command changes the private IP address of the virtual machine to static:

```
## Place virtual network configuration into a variable. ##
$net = @{
    Name = 'myVM'
    ResourceGroupName = 'myResourceGroup'
}
$vnet = Get-AzVirtualNetwork @net

## Place subnet configuration into a variable. ##
$sub = @{
    Name = 'myVM'
    VirtualNetwork = $vnet
}
$subnet = Get-AzVirtualNetworkSubnetConfig @sub

## Get name of network interface and place into a variable ##
$int1 = @{
    Name = 'myVM'
    ResourceGroupName = 'myResourceGroup'
}
$vm = Get-AzVM @int1

## Place network interface configuration into a variable. ##
$nic = Get-AzNetworkInterface -ResourceId $vm.NetworkProfile.NetworkInterfaces.Id

## Set interface configuration. ##
$config = @{
    Name = 'myVM'
    PrivateIpAddress = '192.168.1.4'
    Subnet = $subnet
}
$nic | Set-AzNetworkInterfaceIpConfig @config -Primary

## Save interface configuration. ##
$nic | Set-AzNetworkInterface
```

#### WARNING

Though you can add private IP address settings to the operating system, we recommend not doing so until after reading [Add a private IP address to an operating system](#).

## Clean up resources

When no longer needed, you can use [Remove-AzResourceGroup](#) to remove the resource group and all of the resources it contains:

```
Remove-AzResourceGroup -Name myResourceGroup -Force
```

## Next steps

- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.
- Learn more about creating [Linux](#) and [Windows](#) virtual machines.

# Create a virtual machine with a static private IP address using the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

A virtual machine (VM) is automatically assigned a private IP address from a range that you specify. This range is based on the subnet in which the VM is deployed. The VM keeps the address until the VM is deleted. Azure dynamically assigns the next available private IP address from the subnet you create a VM in. Assign a static IP address to the VM if you want a specific IP address in the subnet.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- An Azure account with an active subscription. [Create an account for free](#).
- This tutorial requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with `az group create` named `myResourceGroup` in the `eastus2` location.

```
az group create \
--name myResourceGroup \
--location eastus2
```

## Create a virtual machine

Create a virtual machine with `az vm create`.

The following command creates a Windows Server virtual machine. When prompted, provide a username and password to be used as the credentials for the virtual machine:

```
az vm create \
--name myVM \
--resource-group myResourceGroup \
--public-ip-address myPublicIP \
--public-ip-sku Standard \
--size Standard_A2 \
--image MicrosoftWindowsServer:WindowsServer:2019-Datacenter:latest \
--admin-username azureuser
```

## Change private IP address to static

In this section, you'll change the private IP address from **dynamic** to **static** for the virtual machine you created previously.

Use [az network nic ip-config update](#) to update the network interface configuration.

The following command changes the private IP address of the virtual machine to static:

```
az network nic ip-config update \
--name ipconfigmyVM \
--resource-group myResourceGroup \
--nic-name myVMVNNic \
--private-ip-address 10.0.0.4
```

### WARNING

Though you can add private IP address settings to the operating system, we recommend not doing so until after reading [Add a private IP address to an operating system](#).

## Clean up resources

When no longer needed, you can use [az group delete](#) to remove the resource group and all of the resources it contains:

```
az group delete --name myResourceGroup --yes
```

## Next steps

- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.
- Learn more about creating [Linux](#) and [Windows](#) virtual machines.

# Assign multiple IP addresses to virtual machines using the Azure portal

9/21/2022 • 16 minutes to read • [Edit Online](#)

An Azure Virtual Machine (VM) has one or more network interfaces (NIC) attached to it. Any NIC can have one or more static or dynamic public and private IP addresses assigned to it.

Assigning multiple IP addresses to a VM enables the following capabilities:

- Hosting multiple websites or services with different IP addresses and TLS/SSL certificates on a single server.
- Serve as a network virtual appliance, such as a firewall or load balancer.
- The ability to add any of the private IP addresses for any of the NICs to an Azure Load Balancer back-end pool. In the past, only the primary IP address for the primary NIC could be added to a back-end pool. For more information about load balancing multiple IP configurations, see [Load balancing multiple IP configurations](#).

Every NIC attached to a VM has one or more IP configurations associated to it. Each configuration is assigned one static or dynamic private IP address. Each configuration may also have one public IP address resource associated to it. To learn more about IP addresses in Azure, read the [IP addresses in Azure](#) article.

## NOTE

All IP configurations on a single NIC must be associated to the same subnet. If multiple IPs on different subnets are desired, multiple NICs on a VM can be used. To learn more about multiple NICs on a VM in Azure, read the [Create VM with Multiple NICs](#) article.

There's a limit to how many private IP addresses can be assigned to a NIC. There's also a limit to how many public IP addresses that can be used in an Azure subscription. See the [Azure limits](#) article for details.

This article explains how to add multiple IP addresses to a virtual machine using the Azure portal.

## NOTE

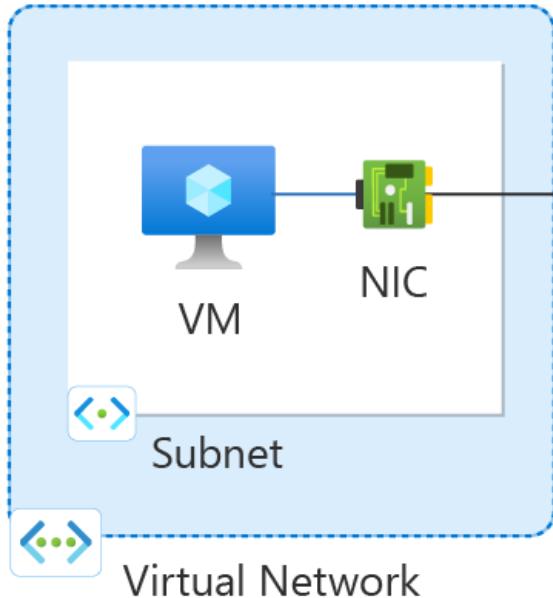
If you want to create a virtual machine with multiple IP addresses, or a static private IP address, you must create it using [PowerShell](#) or the [Azure CLI](#).

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- An existing Azure virtual machine. For more information about creating a virtual machine, see [Create a Windows VM](#) or [Create a Linux VM](#).
  - The example used in this article is named **myVM**. Replace this value with your virtual machine name.

**NOTE**

Though the steps in this article assigns all IP configurations to a single NIC, you can also assign multiple IP configurations to any NIC in a multi-NIC VM. To learn how to create a VM with multiple NICs, see [Create a VM with multiple NICs](#).

**ipconfig1**

- Private IP address: Dynamic
- Public IP address: Static

**ipconfig2**

- Private IP address: Static
- Public IP address: Static

**ipconfig3**

- Private IP address: Static

Figure: Diagram of network configuration resources cerated in How-to article.

## Add public and private IP address to a VM

You can add a private and public IP address to an Azure network interface by completing the following steps.

1. Sign in to the [Azure portal](#).
2. In the search box at the top of the portal, enter **Virtual machine**. Select **Virtual machines** in the search results.
3. In **Virtual machines**, select **myVM** or the name of your virtual machine.
4. Select **Networking** in **Settings**.
5. Select the name of the network interface of the virtual machine. In this example, it's named **myvm889\_z1**.

myVM | Networking

Virtual machine

Search (Ctrl+ /) < Attach network interface Detach network interface Feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings Networking

**myvm889\_z1**

IP configuration ipconfig1 (Primary)

**Network Interface myvm889\_z1** Effective security rules Troubleshoot VM connection issues

Topology Virtual network/subnet: myVNet/myBackendSubnet NIC Public IP: 20.246.73.15 NIC Private IP: 10.1.0.4 Accelerated networking: Disabled

Inbound port rules Outbound port rules Application security groups Load balancing

6. In the network interface, select **IP configurations** in **Settings**.
7. The existing IP configuration is displayed. This configuration is created when the virtual machine is created. To add a private and public IP address to the virtual machine, select **+ Add**.
8. In **Add IP configuration**, enter or select the following information.

SETTING	VALUE
Name	Enter ipconfig2.
Private IP address settings	
Allocation	Select <b>Static</b> .
IP address	Enter an unused address in the network for your virtual machine. For the 10.1.0.0/24 subnet in the example, an IP would be <b>10.1.0.5</b> .
Public IP address	Select <b>Associate</b>
Public IP address	Select <b>Create new</b> . Enter <b>myPublicIP-2</b> in Name. Select <b>Standard</b> in SKU. Select <b>OK</b> .

9. Select **OK**.

# Add IP configuration

X

myvm889\_z1

Name \*

ipconfig2



IP version

IPv4  IPv6

Type

Primary  Secondary

Primary IP configuration already exists

Private IP address settings

Allocation

Dynamic  Static

IP address \*

10.1.0.5



Public IP address

Disassociate  Associate

Public IP address \*

(New) myPublicIP-2



[Create new](#)

**OK**

## NOTE

When adding a static IP address, you must specify an unused, valid address on the subnet the NIC is connected to. If the address you select is not available, the portal displays an X for the IP address and you must select a different one.

## IMPORTANT

After you change the IP address configuration, you must restart the VM for the changes to take effect in the VM.

## Add private IP address to a virtual machine

You can add a private IP address to a virtual machine by completing the following steps.

1. Sign in to the [Azure portal](#).

2. In the search box at the top of the portal, enter **Virtual machine**. Select **Virtual machines** in the search results.
3. In **Virtual machines**, select **myVM** or the name of your virtual machine.
4. Select **Networking** in **Settings**.
5. Select the name of the network interface of the virtual machine. In this example, it's named **myvm889\_z1**.

**myVM | Networking**

Virtual machine

Search (Ctrl+ /)

Attach network interface Detach network interface Feedback

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Settings Networking

myvm889\_z1

IP configuration: ipconfig1 (Primary)

Network Interface: myvm889\_z1 (Topology)

Effective security rules Troubleshoot VM connection issues

Virtual network/subnet: myVNet/myBackendSubnet NIC Public IP: 20.246.73.15 NIC Private IP: 10.1.0.4  
Accelerated networking: Disabled

Inbound port rules Outbound port rules Application security groups Load balancing

6. In the network interface, select **IP configurations** in **Settings**.
7. The existing IP configuration is displayed. This configuration is created when the virtual machine is created. To add a private and public IP address to the virtual machine, select **+ Add**.
8. In **Add IP configuration**, enter or select the following information.

SETTING	VALUE
Name	Enter ipconfig3.
Private IP address settings	
Allocation	Select <b>Static</b> .
IP address	Enter an unused address in the network for your virtual machine. For the 10.1.0.0/24 subnet in the example, an IP would be <b>10.1.0.6</b> .

9. Select **OK**.

# Add IP configuration

X

myvm889\_z1

Name \*

ipconfig3



IP version

IPv4  IPv6

Type

Primary Secondary

Primary IP configuration already exists

Private IP address settings

Allocation

Dynamic Static

IP address \*

10.1.0.6



Public IP address

Disassociate Associate

OK

## NOTE

When adding a static IP address, you must specify an unused, valid address on the subnet the NIC is connected to. If the address you select is not available, the portal displays an X for the IP address and you must select a different one.

## IMPORTANT

After you change the IP address configuration, you must restart the VM for the changes to take effect in the VM.

## Add IP addresses to a VM operating system

Connect and sign in to a VM you created with multiple private IP addresses. You must manually add all the private IP addresses, including the primary, that you added to the VM. Complete the following steps for your VM

operating system.

#### **Windows Server**

- ▶ Expand

#### **Linux (Ubuntu 14/16)**

- ▶ Expand

#### **Linux (Ubuntu 18.04+)**

- ▶ Expand

#### **Linux (Red Hat, CentOS, and others)**

- ▶ Expand

#### **Debian GNU/Linux**

- ▶ Expand

# Assign multiple IP addresses to virtual machines using Azure PowerShell

9/21/2022 • 18 minutes to read • [Edit Online](#)

An Azure Virtual Machine (VM) has one or more network interfaces (NIC) attached to it. Any NIC can have one or more static or dynamic public and private IP addresses assigned to it.

Assigning multiple IP addresses to a VM enables the following capabilities:

- Hosting multiple websites or services with different IP addresses and TLS/SSL certificates on a single server.
- Serve as a network virtual appliance, such as a firewall or load balancer.
- The ability to add any of the private IP addresses for any of the NICs to an Azure Load Balancer back-end pool. In the past, only the primary IP address for the primary NIC could be added to a back-end pool. For more information about load balancing multiple IP configurations, see [Load balancing multiple IP configurations](#).

Every NIC attached to a VM has one or more IP configurations associated to it. Each configuration is assigned one static or dynamic private IP address. Each configuration may also have one public IP address resource associated to it. To learn more about IP addresses in Azure, read the [IP addresses in Azure](#) article.

## NOTE

All IP configurations on a single NIC must be associated to the same subnet. If multiple IPs on different subnets are desired, multiple NICs on a VM can be used. To learn more about multiple NICs on a VM in Azure, read the [Create VM with Multiple NICs](#) article.

There's a limit to how many private IP addresses can be assigned to a NIC. There's also a limit to how many public IP addresses that can be used in an Azure subscription. See the [Azure limits](#) article for details.

This article explains how to add multiple IP addresses to a virtual machine using the Azure portal.

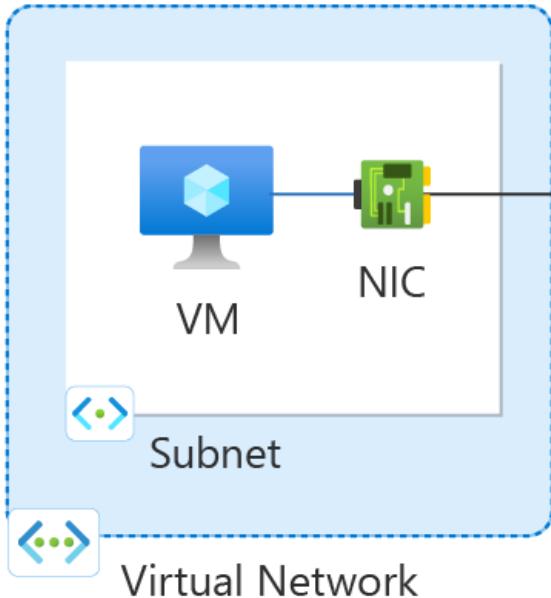
## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- Azure PowerShell installed locally or Azure Cloud Shell.
- Sign in to Azure PowerShell and ensure you've selected the subscription with which you want to use this feature. For more information, see [Sign in with Azure PowerShell](#).
- Ensure your Az. Network module is 4.3.0 or later. To verify the installed module, use the command `Get-InstalledModule -Name "Az.Network"`. If the module requires an update, use the command `Update-Module -Name "Az. Network"` if necessary.

If you choose to install and use PowerShell locally, this article requires the Azure PowerShell module version 5.4.1 or later. Run `Get-Module -ListAvailable Az` to find the installed version. If you need to upgrade, see [Install Azure PowerShell module](#). If you're running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

#### NOTE

Though the steps in this article assigns all IP configurations to a single NIC, you can also assign multiple IP configurations to any NIC in a multi-NIC VM. To learn how to create a VM with multiple NICs, see [Create a VM with multiple NICs](#).



#### ipconfig1

- Private IP address: Dynamic
- Public IP address: Static

#### ipconfig2

- Private IP address: Static
- Public IP address: Static

#### ipconfig3

- Private IP address: Static

Figure: Diagram of network configuration resources created in How-to article.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [New-AzResourceGroup](#) named **myResourceGroup** in the **eastus2** location.

```
$rg =@{  
    Name = 'myResourceGroup'  
    Location = 'eastus2'  
}  
New-AzResourceGroup @rg
```

## Create a virtual network

In this section, you'll create a virtual network for the virtual machine.

Use [New-AzVirtualNetwork](#) and [New-AzVirtualNetworkSubnetConfig](#) to create a virtual network.

```
## Create backend subnet config ##
$subnet = @{
    Name = 'myBackendSubnet'
    AddressPrefix = '10.1.0.0/24'
}
$subnetConfig = New-AzVirtualNetworkSubnetConfig @subnet

## Create the virtual network ##
$net = @{
    Name = 'myVNet'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    AddressPrefix = '10.1.0.0/16'
    Subnet = $subnetConfig
}
New-AzVirtualNetwork @net
```

## Create primary public IP address

Use [New-AzPublicIpAddress](#) to create a primary public IP address.

```
$ip1 = @{
    Name = 'myPublicIP-1'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    Sku = 'Standard'
    AllocationMethod = 'Static'
    ipAddressVersion = 'IPv4'
    Zone = 1,2,3
}
New-AzPublicIpAddress @ip1
```

## Create a network security group

In this section, you'll create a network security group for the virtual machine and virtual network. You'll create a rule to allow connections to the virtual machine on port 22 for SSH.

Use [New-AzNetworkSecurityGroup](#) and [New-AzNetworkSecurityRuleConfig](#) to create the network security group and rules.

```

## Create rule for network security group and place in variable. ##
$nsgrule1 = @{
    Name = 'myNSGRuleSSH'
    Description = 'Allow SSH'
    Protocol = '*'
    SourcePortRange = '*'
    DestinationPortRange = '22'
    SourceAddressPrefix = 'Internet'
    DestinationAddressPrefix = '*'
    Access = 'Allow'
    Priority = '200'
    Direction = 'Inbound'
}
$rule1 = New-AzNetworkSecurityRuleConfig @nsgrule1

## Create network security group ##
$nsg = @{
    Name = 'myNSG'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    SecurityRules = $rule1
}
New-AzNetworkSecurityGroup @nsg

```

## Create network interface

You'll use [New-AzNetworkInterface](#) and [New-AzNetworkInterfaceIpConfig](#) to create the network interface for the virtual machine. The public IP addresses and the NSG created previously are associated with the NIC. The network interface is attached to the virtual network you created previously.

```

## Place the virtual network into a variable. ##
$net = @{
    Name = 'myVNet'
    ResourceGroupName = 'myResourceGroup'
}
$vnet = Get-AzVirtualNetwork @net

## Place the network security group into a variable. ##
$ns = @{
    Name = 'myNSG'
    ResourceGroupName = 'myResourceGroup'
}
$nsg = Get-AzNetworkSecurityGroup @ns

## Place the primary public IP address into a variable. ##
$pub1 = @{
    Name = 'myPublicIP-1'
    ResourceGroupName = 'myResourceGroup'
}
$pubIP1 = Get-AzPublicIPAddress @pub1

## Create primary configuration for NIC. ##
$IP1 = @{
    Name = 'ipconfig1'
    Subnet = $vnet.Subnets[0]
    PrivateIpAddressVersion = 'IPv4'
    PublicIPAddress = $pubIP1
}
$IP1Config = New-AzNetworkInterfaceIpConfig @IP1 -Primary

## Create tertiary configuration for NIC. ##
$IP3 = @{
    Name = 'ipconfig3'
    Subnet = $vnet.Subnets[0]
    PrivateIpAddressVersion = 'IPv4'
    PrivateIpAddress = '10.1.0.6'
}
$IP3Config = New-AzNetworkInterfaceIpConfig @IP3

## Command to create network interface for VM ##
$nic = @{
    Name = 'myNIC1'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    NetworkSecurityGroup = $nsg
    IpConfiguration = $IP1Config,$IP3Config
}
New-AzNetworkInterface @nic

```

#### **NOTE**

When adding a static IP address, you must specify an unused, valid address on the subnet the NIC is connected to.

## **Create virtual machine**

Use the following commands to create the virtual machine:

- [New-AzVM](#)
- [New-AzVMConfig](#)
- [Set-AzVMOperatingSystem](#)
- [Set-AzVMSourceImage](#)

- [Add-AzVMNetworkInterface](#)

```
$cred = Get-Credential

## Place network interface into a variable. ##
$nic = @{
    Name = 'myNIC1'
    ResourceGroupName = 'myResourceGroup'
}
$nicVM = Get-AzNetworkInterface @nic

## Create a virtual machine configuration for VMs ##
$vmsz = @{
    VMName = 'myVM'
    VMSize = 'Standard_DS1_v2'
}
$vmos = @{
    ComputerName = 'myVM'
    Credential = $cred
}
$vmimage = @{
    PublisherName = 'Debian'
    Offer = 'debian-11'
    Skus = '11'
    Version = 'latest'
}
$vmConfig = New-AzVMConfig @vmsz `

| Set-AzVMOperatingSystem @vmos -Linux `

| Set-AzVMSourceImage @vmimage `

| Add-AzVMNetworkInterface -Id $nicVM.Id

## Create the virtual machine for VMs ##
$vm = @{
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    VM = $vmConfig
    SshKeyName = 'mySSHKey'
}
New-AzVM @vm -GenerateSshKey
```

## Add secondary private and public IP address

Use [New-AzPublicIpAddress](#) to create a secondary public IP address.

```
$ip2 = @{
    Name = 'myPublicIP-2'
    ResourceGroupName = 'myResourceGroup'
    Location = 'eastus2'
    Sku = 'Standard'
    AllocationMethod = 'Static'
    IpAddressVersion = 'IPv4'
    Zone = 1,2,3
}
New-AzPublicIpAddress @ip2
```

Use [New-AzNetworkInterfaceIpConfig](#) to create the secondary IP configuration for the virtual machine.

```

## Place the virtual network into a variable. ##
$net = @{
    Name = 'myVNet'
    ResourceGroupName = 'myResourceGroup'
}
$vnet = Get-AzVirtualNetwork @net

## Place your virtual network subnet into a variable. ##
$sub = @{
    Name = 'myBackendSubnet'
    VirtualNetwork = $vnet
}
$subnet = Get-AzVirtualNetworkSubnetConfig @sub

## Place the secondary public IP address you created previously into a variable. ##
$pip = @{
    Name = 'myPublicIP-2'
    ResourceGroupName = 'myResourceGroup'
}
$pubIP2 = Get-AzPublicIPAddress @pip

## Place the network interface into a variable. ##
$net = @{
    Name = 'myNIC1'
    ResourceGroupName = 'myResourceGroup'
}
$nic = Get-AzNetworkInterface @net

## Create secondary configuration for NIC. ##
$IPc2 = @{
    Name = 'ipconfig2'
    Subnet = $vnet.Subnets[0]
    PrivateIpAddressVersion = 'IPv4'
    PrivateIpAddress = '10.1.0.5'
    PublicIPAddress = $pubIP2
}
$IP2Config = New-AzNetworkInterfaceIpConfig @IPc2

## Add the IP configuration to the network interface. ##
$nic.IpConfigurations.Add($IP2Config)

## Save the configuration to the network interface. ##
$nic | Set-AzNetworkInterface

```

## Add IP addresses to a VM operating system

Connect and sign in to a VM you created with multiple private IP addresses. You must manually add all the private IP addresses, including the primary, that you added to the VM. Complete the following steps for your VM operating system.

### **Windows Server**

- ▶ Expand

### **Linux (Ubuntu 14/16)**

- ▶ Expand

### **Linux (Ubuntu 18.04+)**

- ▶ Expand

### **Linux (Red Hat, CentOS, and others)**

- ▶ Expand

## **Debian GNU/Linux**

► Expand

# Assign multiple IP addresses to virtual machines using the Azure CLI

9/21/2022 • 17 minutes to read • [Edit Online](#)

An Azure Virtual Machine (VM) has one or more network interfaces (NIC) attached to it. Any NIC can have one or more static or dynamic public and private IP addresses assigned to it.

Assigning multiple IP addresses to a VM enables the following capabilities:

- Hosting multiple websites or services with different IP addresses and TLS/SSL certificates on a single server.
- Serve as a network virtual appliance, such as a firewall or load balancer.
- The ability to add any of the private IP addresses for any of the NICs to an Azure Load Balancer back-end pool. In the past, only the primary IP address for the primary NIC could be added to a back-end pool. For more information about load balancing multiple IP configurations, see [Load balancing multiple IP configurations](#).

Every NIC attached to a VM has one or more IP configurations associated to it. Each configuration is assigned one static or dynamic private IP address. Each configuration may also have one public IP address resource associated to it. To learn more about IP addresses in Azure, read the [IP addresses in Azure](#) article.

## NOTE

All IP configurations on a single NIC must be associated to the same subnet. If multiple IPs on different subnets are desired, multiple NICs on a VM can be used. To learn more about multiple NICs on a VM in Azure, read the [Create VM with Multiple NICs](#) article.

There's a limit to how many private IP addresses can be assigned to a NIC. There's also a limit to how many public IP addresses that can be used in an Azure subscription. See the [Azure limits](#) article for details.

This article explains how to add multiple IP addresses to a virtual machine using the Azure portal.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.



[Launch Cloud Shell](#)

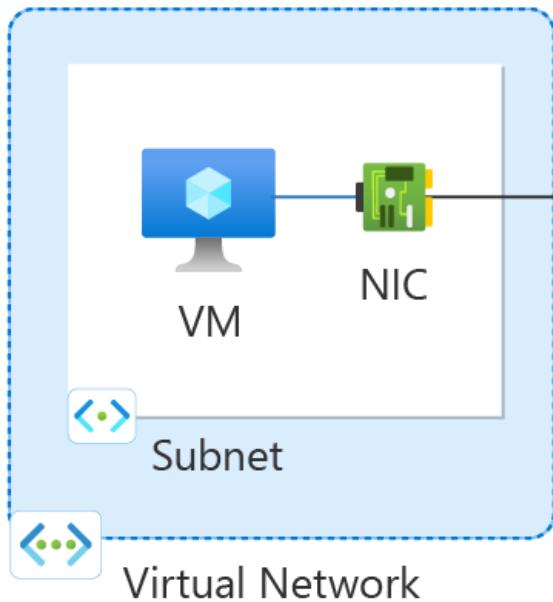
- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about

extensions, see [Use extensions with the Azure CLI](#).

- Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This tutorial requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

#### NOTE

Though the steps in this article assigns all IP configurations to a single NIC, you can also assign multiple IP configurations to any NIC in a multi-NIC VM. To learn how to create a VM with multiple NICs, see [Create a VM with multiple NICs](#).



#### ipconfig1

- Private IP address: Dynamic
- Public IP address: Static

#### ipconfig2

- Private IP address: Static
- Public IP address: Static

#### ipconfig3

- Private IP address: Static

Figure: Diagram of network configuration resources created in How-to article.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with `az group create` named `myResourceGroup` in the `eastus2` location.

```
az group create \
--name myResourceGroup \
--location eastus2
```

## Create a virtual network

In this section, you'll create a virtual network for the virtual machine.

Use `az network vnet create` to create a virtual network.

```
az network vnet create \
--resource-group myResourceGroup \
--location eastus2 \
--name myVNet \
--address-prefixes 10.1.0.0/16 \
--subnet-name myBackendSubnet \
--subnet-prefixes 10.1.0.0/24
```

## Create public IP addresses

Use [az network public-ip create](#) to create two public IP addresses.

```
az network public-ip create \
--resource-group myResourceGroup \
--name myPublicIP-1 \
--sku Standard \
--version IPv4 \
--zone 1 2 3

az network public-ip create \
--resource-group myResourceGroup \
--name myPublicIP-2 \
--sku Standard \
--version IPv4 \
--zone 1 2 3
```

## Create a network security group

In this section, you'll create a network security group for the virtual machine and virtual network.

Use [az network nsg create](#) to create the network security group.

```
az network nsg create \
--resource-group myResourceGroup \
--name myNSG
```

### Create network security group rules

You'll create a rule to allow connections to the virtual machine on port 22 for SSH.

Use [az network nsg rule create](#) to create the network security group rules.

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNSG \
--name myNSGRuleSSH \
--protocol '*' \
--direction inbound \
--source-address-prefix '*' \
--source-port-range '*' \
--destination-address-prefix '*' \
--destination-port-range 22 \
--access allow \
--priority 200
```

## Create network interface

You'll use [az network nic create](#) to create the network interface for the virtual machine. The public IP addresses and the NSG created previously are associated with the NIC. The network interface is attached to the virtual network you created previously.

```
az network nic create \
--resource-group myResourceGroup \
--name myNIC1 \
--private-ip-address-version IPv4 \
--vnet-name myVNet \
--subnet myBackendSubnet \
--network-security-group myNSG \
--public-ip-address myPublicIP-1
```

### Create secondary private and public IP configuration

Use [az network nic ip-config create](#) to create the secondary private and public IP configuration for the NIC.

Replace 10.1.0.5 with your secondary private IP address.

```
az network nic ip-config create \
--resource-group myResourceGroup \
--name ipconfig2 \
--nic-name myNIC1 \
--private-ip-address 10.1.0.5 \
--private-ip-address-version IPv4 \
--vnet-name myVNet \
--subnet myBackendSubnet \
--public-ip-address myPublicIP-2
```

### Create tertiary private IP configuration

Use [az network nic ip-config create](#) to create the tertiary private IP configuration for the NIC. Replace 10.1.0.6 with your secondary private IP address.

```
az network nic ip-config create \
--resource-group myResourceGroup \
--name ipconfig3 \
--nic-name myNIC1 \
--private-ip-address 10.1.0.6 \
--private-ip-address-version IPv4 \
--vnet-name myVNet \
--subnet myBackendSubnet
```

#### NOTE

When adding a static IP address, you must specify an unused, valid address on the subnet the NIC is connected to.

### Create virtual machine

Use [az vm create](#) to create the virtual machine.

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--nics myNIC1 \
--image UbuntuLTS \
--admin-username azureuser \
--authentication-type ssh \
--generate-ssh-keys
```

## Add IP addresses to a VM operating system

Connect and sign in to a VM you created with multiple private IP addresses. You must manually add all the private IP addresses, including the primary, that you added to the VM. Complete the following steps for your VM operating system.

### **Windows Server**

▶ Expand

### **Linux (Ubuntu 14/16)**

▶ Expand

### **Linux (Ubuntu 18.04+)**

▶ Expand

### **Linux (Red Hat, CentOS, and others)**

▶ Expand

### **Debian GNU/Linux**

▶ Expand

# Add network interfaces to or remove network interfaces from virtual machines

9/21/2022 • 8 minutes to read • [Edit Online](#)

Learn how to add an existing network interface when you create an Azure virtual machine (VM). Also learn to add or remove network interfaces from an existing VM in the stopped (deallocated) state. A network interface enables an Azure VM to communicate with internet, Azure, and on-premises resources. A VM has one or more network interfaces.

If you need to add, change, or remove IP addresses for a network interface, see [Manage network interface IP addresses](#). To create, change, or delete network interfaces, see [Manage network interfaces](#).

## Before you begin

### NOTE

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az](#).

If you don't have one, set up an Azure account with an active subscription. [Create an account for free](#). Complete one of these tasks before starting the remainder of this article:

- **Portal users:** Sign in to the [Azure portal](#) with your Azure account.
- **PowerShell users:** Either run the commands in the [Azure Cloud Shell](#), or run PowerShell from your computer. The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account. In the Azure Cloud Shell browser tab, find the **Select environment** dropdown list, then pick **PowerShell** if it isn't already selected.

If you're running PowerShell locally, use Azure PowerShell module version 1.0.0 or later. Run `Get-Module -ListAvailable Az.Network` to find the installed version. If you need to upgrade, see [Install Azure PowerShell module](#). Run `Connect-AzAccount` to create a connection with Azure.

- **Azure CLI users:** Run the commands via either the [Azure Cloud Shell](#) or the Azure CLI running locally. Use Azure CLI version 2.0.26 or later if you're running the Azure CLI locally. Run `az --version` to find the installed version. If you need to install or upgrade, see [Install Azure CLI](#). Run `az login` to create a connection with Azure.

## Add existing network interfaces to a new VM

When you create a virtual machine through the portal, the portal creates a network interface with default settings and attaches the network interface to the VM for you. You can't use the portal to add existing network interfaces to a new VM, or to create a VM with multiple network interfaces. You can do both by using the CLI or PowerShell. Be sure to familiarize yourself with the [constraints](#). If you create a VM with multiple network interfaces, you must also configure the operating system to use them properly after you create the VM. Learn how to configure [Linux](#) or [Windows](#) for multiple network interfaces.

### Commands

Before you create the VM, [Create a network interface](#).

TOOL	COMMAND
CLI	<a href="#">az network nic create</a>
PowerShell	<a href="#">New-AzNetworkInterface</a>

## Add a network interface to an existing VM

To add a network interface to your virtual machine:

1. Go to the [Azure portal](#) to find an existing virtual machine. Search for and select **Virtual machines**.
2. Select the name of your VM. The VM must support the number of network interfaces you want to add. To find out how many network interfaces each VM size supports, see the sizes in Azure for [Linux VMs](#) or [Windows VMs](#).
3. In the VM command bar, select **Stop**, and then **OK** in the confirmation dialog box. Then wait until the **Status** of the VM changes to **Stopped (deallocated)**.
4. From the VM menu bar, choose **Networking > Attach network interface**. Then in **Attach existing network interface**, choose the network interface you'd like to attach, and select **OK**.

### NOTE

The network interface you select can't have accelerated networking enabled, can't have an IPv6 address assigned to it, and must exist in the same virtual network with the network interface currently attached to the VM.

If you don't have an existing network interface, you must first create one. To do so, select **Create network interface**. To learn more about how to create a network interface, see [Create a network interface](#). To learn more about additional constraints when adding network interfaces to virtual machines, see [Constraints](#).

5. From the VM menu bar, choose **Overview > Start** to restart the virtual machine.

Now you can configure the VM operating system to use multiple network interfaces properly. Learn how to configure [Linux](#) or [Windows](#) for multiple network interfaces.

## Commands

TOOL	COMMAND
CLI	<a href="#">az vm nic add</a> (reference); <a href="#">detailed steps</a>
PowerShell	<a href="#">Add-AzVMNetworkInterface</a> (reference); <a href="#">detailed steps</a>

## View network interfaces for a VM

You can view the network interfaces currently attached to a VM to learn about each network interface's configuration, and the IP addresses assigned to each network interface.

1. Go to the [Azure portal](#) to find an existing virtual machine. Search for and select **Virtual machines**.

**NOTE**

Sign in using an account that is assigned the Owner, Contributor, or Network Contributor role for your subscription. To learn more about how to assign roles to accounts, see [Built-in roles for Azure role-based access control](#).

2. Select the name of the VM for which you want to view attached network interfaces.

3. In the VM menu bar, select **Networking**.

To learn about network interface settings and how to change them, see [Manage network interfaces](#). To learn about how to add, change, or remove IP addresses assigned to a network interface, see [Manage network interface IP addresses](#).

**Commands**

TOOL	COMMAND
CLI	<code>az vm nic list</code>
PowerShell	<code>Get-AzVM</code>

## Remove a network interface from a VM

1. Go to the [Azure portal](#) to find an existing virtual machine. Search for and select **Virtual machines**.
2. Select the name of the VM for which you want to view attached network interfaces.
3. In the VM toolbar, pick **Stop**.
4. Wait until the **Status** of the VM changes to **Stopped (deallocated)**.
5. From the VM menu bar, choose **Networking > Detach network interface**.
6. In the **Detach network interface** dialog box, select the network interface you'd like to detach. Then select **OK**.

**NOTE**

If only one network interface is listed, you can't detach it, because a virtual machine must always have at least one network interface attached to it.

**Commands**

TOOL	COMMAND
CLI	<code>az vm nic remove</code> (reference); <a href="#">detailed steps</a>
PowerShell	<code>Remove-AzVMNetworkInterface</code> (reference); <a href="#">detailed steps</a>

## Constraints

- A VM must have at least one network interface attached to it.
- A VM can only have as many network interfaces attached to it as the VM size supports. To learn more about how many network interfaces each VM size supports, see the sizes in Azure for [Linux VMs](#) or

[Windows VMs](#). All sizes support at least two network interfaces.

- The network interfaces you add to a VM can't currently be attached to another VM. To learn more about how to create network interfaces, see [Create a network interface](#).
- In the past, you could add network interfaces only to VMs that supported multiple network interfaces and were created with at least two network interfaces. You couldn't add a network interface to a VM that was created with one network interface, even if the VM size supported more than one network interface. Conversely, you could only remove network interfaces from a VM with at least three network interfaces, because VMs created with at least two network interfaces always had to have at least two network interfaces. These constraints no longer apply. You can now create a VM with any number of network interfaces (up to the number supported by the VM size).
- By default, the first network interface attached to a VM is the *primary* network interface. All other network interfaces in the VM are *secondary* network interfaces.
- You can control which network interface you send outbound traffic to. However, a VM by default sends all outbound traffic to the IP address that's assigned to the primary IP configuration of the primary network interface.
- In the past, all VMs within the same availability set were required to have a single, or multiple, network interfaces. VMs with any number of network interfaces can now exist in the same availability set, up to the number supported by the VM size. You can only add a VM to an availability set when it's created. To learn more about availability sets, see [Manage the availability of VMs in Azure](#).
- You can connect network interfaces in the same VM to different subnets within a virtual network. However, the network interfaces must all be connected to the same virtual network.
- You can add any IP address for any IP configuration of any primary or secondary network interface to an Azure Load Balancer back-end pool. In the past, only the primary IP address for the primary network interface could be added to a back-end pool. To learn more about IP addresses and configurations, see [Add, change, or remove IP addresses](#).
- Deleting a VM doesn't delete the network interfaces that are attached to it. When you delete a VM, the network interfaces are detached from the VM. You can add those network interfaces to different VMs or delete them.
- Achieving the optimal performance documented requires Accelerated Networking. In some cases, you must explicitly enable Accelerated Networking for [Windows](#) or [Linux](#) virtual machines.

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Next steps

To create a VM with multiple network interfaces or IP addresses, see:

TASK	TOOL
Create a VM with multiple NICs	<a href="#">CLI</a> , <a href="#">PowerShell</a>
Create a single NIC VM with multiple IPv4 addresses	<a href="#">CLI</a> , <a href="#">PowerShell</a>
Create a single NIC VM with a private IPv6 address (behind an Azure Load Balancer)	<a href="#">CLI</a> , <a href="#">PowerShell</a> , <a href="#">Azure Resource Manager template</a>

# Create and manage a Windows virtual machine that has multiple NICs

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Virtual machines (VMs) in Azure can have multiple virtual network interface cards (NICs) attached to them. A common scenario is to have different subnets for front-end and back-end connectivity. You can associate multiple NICs on a VM to multiple subnets, but those subnets must all reside in the same virtual network (vNet). This article details how to create a VM that has multiple NICs attached to it. You also learn how to add or remove NICs from an existing VM. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly.

## NOTE

If multiple subnets are not required for a scenario, it may be more straightforward to utilize multiple IP configurations on a single NIC. Instructions for this setup can be found [here](#).

## Prerequisites

In the following examples, replace example parameter names with your own values. Example parameter names include *myResourceGroup*, *myVnet*, and *myVM*.

## Create a VM with multiple NICs

First, create a resource group. The following example creates a resource group named *myResourceGroup* in the *EastUs* location:

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

### Create virtual network and subnets

A common scenario is for a virtual network to have two or more subnets. One subnet may be for front-end traffic, the other for back-end traffic. To connect to both subnets, you then use multiple NICs on your VM.

1. Define two virtual network subnets with [New-AzVirtualNetworkSubnetConfig](#). The following example defines the subnets for *mySubnetFrontEnd* and *mySubnetBackEnd*.

```
$mySubnetFrontEnd = New-AzVirtualNetworkSubnetConfig -Name "mySubnetFrontEnd" `  
    -AddressPrefix "192.168.1.0/24"  
$mySubnetBackEnd = New-AzVirtualNetworkSubnetConfig -Name "mySubnetBackEnd" `  
    -AddressPrefix "192.168.2.0/24"
```

2. Create your virtual network and subnets with [New-AzVirtualNetwork](#). The following example creates a virtual network named *myVnet*.

```
$myVnet = New-AzVirtualNetwork -ResourceGroupName "myResourceGroup" ` 
    -Location "EastUs" ` 
    -Name "myVnet" ` 
    -AddressPrefix "192.168.0.0/16" ` 
    -Subnet $mySubnetFrontEnd,$mySubnetBackEnd
```

## Create multiple NICs

Create two NICs with [New-AzNetworkInterface](#). Attach one NIC to the front-end subnet and one NIC to the back-end subnet. The following example creates NICs named *myNic1* and *myNic2*:

```
$frontEnd = $myVnet.Subnets|?{$_._Name -eq 'mySubnetFrontEnd'} 
$myNic1 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" ` 
    -Name "myNic1" ` 
    -Location "EastUs" ` 
    -SubnetId $frontEnd.Id

$backEnd = $myVnet.Subnets|?{$_._Name -eq 'mySubnetBackEnd'} 
$myNic2 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" ` 
    -Name "myNic2" ` 
    -Location "EastUs" ` 
    -SubnetId $backEnd.Id
```

Typically you also create a [network security group](#) to filter network traffic to the VM and a [load balancer](#) to distribute traffic across multiple VMs.

## Create the virtual machine

Now start to build your VM configuration. Each VM size has a limit for the total number of NICs that you can add to a VM. For more information, see [Windows VM sizes](#).

- Set your VM credentials to the `$cred` variable as follows:

```
$cred = Get-Credential
```

- Define your VM with [New-AzVMConfig](#). The following example defines a VM named *myVM* and uses a VM size that supports more than two NICs (*Standard\_DS3\_v2*):

```
$vmConfig = New-AzVMConfig -VMName "myVM" -VMSize "Standard_DS3_v2"
```

- Create the rest of your VM configuration with [Set-AzVMOperatingSystem](#) and [Set-AzVMSourceImage](#). The following example creates a Windows Server 2016 VM:

```
$vmConfig = Set-AzVMOperatingSystem -VM $vmConfig ` 
    -Windows ` 
    -ComputerName "myVM" ` 
    -Credential $cred ` 
    -ProvisionVMAgent ` 
    -EnableAutoUpdate
$vmConfig = Set-AzVMSourceImage -VM $vmConfig ` 
    -PublisherName "MicrosoftWindowsServer" ` 
    -Offer "WindowsServer" ` 
    -Skus "2016-Datacenter" ` 
    -Version "latest"
```

- Attach the two NICs that you previously created with [Add-AzVMNetworkInterface](#):

```
$vmConfig = Add-AzVMNetworkInterface -VM $vmConfig -Id $myNic1.Id -Primary  
$vmConfig = Add-AzVMNetworkInterface -VM $vmConfig -Id $myNic2.Id
```

5. Create your VM with [New-AzVM](#):

```
New-AzVM -VM $vmConfig -ResourceGroupName "myResourceGroup" -Location "EastUs"
```

6. Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Add a NIC to an existing VM

To add a virtual NIC to an existing VM, you deallocate the VM, add the virtual NIC, then start the VM. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly. If needed, you can [resize a VM](#).

1. Deallocate the VM with [Stop-AzVM](#). The following example deallocates the VM named *myVM* in *myResourceGroup*:

```
Stop-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

2. Get the existing configuration of the VM with [Get-AzVm](#). The following example gets information for the VM named *myVM* in *myResourceGroup*:

```
$vm = Get-AzVm -Name "myVM" -ResourceGroupName "myResourceGroup"
```

3. The following example creates a virtual NIC with [New-AzNetworkInterface](#) named *myNic3* that is attached to *mySubnetBackEnd*. The virtual NIC is then attached to the VM named *myVM* in *myResourceGroup* with [Add-AzVMNetworkInterface](#):

```
# Get info for the back end subnet  
$myVnet = Get-AzVirtualNetwork -Name "myVnet" -ResourceGroupName "myResourceGroup"  
$backEnd = $myVnet.Subnets | ?{$_ . Name -eq 'mySubnetBackEnd' }  
  
# Create a virtual NIC  
$myNic3 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" `  
    -Name "myNic3" `  
    -Location "EastUs" `  
    -SubnetId $backEnd.Id  
  
# Get the ID of the new virtual NIC and add to VM  
$nicId = (Get-AzNetworkInterface -ResourceGroupName "myResourceGroup" -Name "MyNic3").Id  
Add-AzVMNetworkInterface -VM $vm -Id $nicId | Update-AzVm -ResourceGroupName "myResourceGroup"
```

### Primary virtual NICs

One of the NICs on a multi-NIC VM needs to be primary. If one of the existing virtual NICs on the VM is already set as primary, you can skip this step. The following example assumes that two virtual NICs are now present on a VM and you wish to add the first NIC ([0]) as the primary:

```

# List existing NICs on the VM and find which one is primary
$vm.NetworkProfile.NetworkInterfaces

# Set NIC 0 to be primary
$vm.NetworkProfile.NetworkInterfaces[0].Primary = $true
$vm.NetworkProfile.NetworkInterfaces[1].Primary = $false

# Update the VM state in Azure
Update-AzVM -VM $vm -ResourceGroupName "myResourceGroup"

```

4. Start the VM with [Start-AzVm](#):

```
Start-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM"
```

5. Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Remove a NIC from an existing VM

To remove a virtual NIC from an existing VM, you deallocate the VM, remove the virtual NIC, then start the VM.

1. Deallocation the VM with [Stop-AzVM](#). The following example deallocated the VM named *myVM* in *myResourceGroup*:

```
Stop-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

2. Get the existing configuration of the VM with [Get-AzVm](#). The following example gets information for the VM named *myVM* in *myResourceGroup*:

```
$vm = Get-AzVm -Name "myVM" -ResourceGroupName "myResourceGroup"
```

3. Get information about the NIC remove with [Get-AzNetworkInterface](#). The following example gets information about *myNic3*:

```

# List existing NICs on the VM if you need to determine NIC name
$vm.NetworkProfile.NetworkInterfaces

$nicId = (Get-AzNetworkInterface -ResourceGroupName "myResourceGroup" -Name "myNic3").Id

```

4. Remove the NIC with [Remove-AzVMNetworkInterface](#) and then update the VM with [Update-AzVm](#). The following example removes *myNic3* as obtained by `$nicId` in the preceding step:

```
Remove-AzVMNetworkInterface -VM $vm -NetworkInterfaceIDs $nicId | ` 
Update-AzVm -ResourceGroupName "myResourceGroup"
```

5. Start the VM with [Start-AzVm](#):

```
Start-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

## Create multiple NICs with templates

Azure Resource Manager templates provide a way to create multiple instances of a resource during deployment,

such as creating multiple NICs. Resource Manager templates use declarative JSON files to define your environment. For more information, see [overview of Azure Resource Manager](#). You can use `copy` to specify the number of instances to create:

```
"copy": {  
    "name": "multiplenics",  
    "count": "[parameters('count')]"  
}
```

For more information, see [creating multiple instances by using `copy`](#).

You can also use `copyIndex()` to append a number to a resource name. You can then create `myNic1`, `MyNic2` and so on. The following code shows an example of appending the index value:

```
"name": "[concat('myNic', copyIndex())]",
```

You can read a complete example of [creating multiple NICs by using Resource Manager templates](#).

Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Configure guest OS for multiple NICs

Azure assigns a default gateway to the first (primary) network interface attached to the virtual machine. Azure does not assign a default gateway to additional (secondary) network interfaces attached to a virtual machine. Therefore, you are unable to communicate with resources outside the subnet that a secondary network interface is in, by default. Secondary network interfaces can, however, communicate with resources outside their subnet, though the steps to enable communication are different for different operating systems.

- From a Windows command prompt, run the `route print` command, which returns output similar to the following output for a virtual machine with two attached network interfaces:

```
=====  
Interface List  
3...00 0d 3a 10 92 ce .....Microsoft Hyper-V Network Adapter #3  
7...00 0d 3a 10 9b 2a .....Microsoft Hyper-V Network Adapter #4  
=====
```

In this example, **Microsoft Hyper-V Network Adapter #4** (interface 7) is the secondary network interface that doesn't have a default gateway assigned to it.

- From a command prompt, run the `ipconfig` command to see which IP address is assigned to the secondary network interface. In this example, 192.168.2.4 is assigned to interface 7. No default gateway address is returned for the secondary network interface.
- To route all traffic destined for addresses outside the subnet of the secondary network interface to the gateway for the subnet, run the following command:

```
route add -p 0.0.0.0 MASK 0.0.0.0 192.168.2.1 METRIC 5015 IF 7
```

The gateway address for the subnet is the first IP address (ending in .1) in the address range defined for the subnet. If you don't want to route all traffic outside the subnet, you could add individual routes to specific destinations, instead. For example, if you only wanted to route traffic from the secondary network interface to the 192.168.3.0 network, you enter the command:

```
route add -p 192.168.3.0 MASK 255.255.255.0 192.168.2.1 METRIC 5015 IF 7
```

4. To confirm successful communication with a resource on the 192.168.3.0 network, for example, enter the following command to ping 192.168.3.4 using interface 7 (192.168.2.4):

```
ping 192.168.3.4 -S 192.168.2.4
```

You may need to open ICMP through the Windows firewall of the device you're pinging with the following command:

```
netsh advfirewall firewall add rule name=Allow-ping protocol=icmpv4 dir=in action=allow
```

5. To confirm the added route is in the route table, enter the `route print` command, which returns output similar to the following text:

```
=====
Active Routes:
Network Destination      Netmask          Gateway        Interface Metric
      0.0.0.0          0.0.0.0    192.168.1.1    192.168.1.4      15
      0.0.0.0          0.0.0.0    192.168.2.1    192.168.2.4    5015
```

The route listed with **192.168.1.1** under **Gateway**, is the route that is there by default for the primary network interface. The route with **192.168.2.1** under **Gateway**, is the route you added.

## Next steps

Review [Windows VM sizes](#) when you're trying to create a VM that has multiple NICs. Pay attention to the maximum number of NICs that each VM size supports.

# How to create a Linux virtual machine in Azure with multiple network interface cards

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article details how to create a VM with multiple NICs with the Azure CLI.

## Create supporting resources

Install the latest [Azure CLI](#) and log in to an Azure account using [az login](#).

In the following examples, replace example parameter names with your own values. Example parameter names included *myResourceGroup*, *mystorageaccount*, and *myVM*.

First, create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name myResourceGroup --location eastus
```

Create the virtual network with [az network vnet create](#). The following example creates a virtual network named *myVnet* and subnet named *mySubnetFrontEnd*.

```
az network vnet create \
--resource-group myResourceGroup \
--name myVnet \
--address-prefix 10.0.0.0/16 \
--subnet-name mySubnetFrontEnd \
--subnet-prefix 10.0.1.0/24
```

Create a subnet for the back-end traffic with [az network vnet subnet create](#). The following example creates a subnet named *mySubnetBackEnd*.

```
az network vnet subnet create \
--resource-group myResourceGroup \
--vnet-name myVnet \
--name mySubnetBackEnd \
--address-prefix 10.0.2.0/24
```

Create a network security group with [az network nsg create](#). The following example creates a network security group named *myNetworkSecurityGroup*.

```
az network nsg create \
--resource-group myResourceGroup \
--name myNetworkSecurityGroup
```

## Create and configure multiple NICs

Create two NICs with [az network nic create](#). The following example creates two NICs, named *myNic1* and *myNic2*, connected the network security group, with one NIC connecting to each subnet:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic1 \
--vnet-name myVnet \
--subnet mySubnetFrontEnd \
--network-security-group myNetworkSecurityGroup
az network nic create \
--resource-group myResourceGroup \
--name myNic2 \
--vnet-name myVnet \
--subnet mySubnetBackEnd \
--network-security-group myNetworkSecurityGroup
```

## Create a VM and attach the NICs

When you create the VM, specify the NICs you created with `--nics`. You also need to take care when you select the VM size. There are limits for the total number of NICs that you can add to a VM. Read more about [Linux VM sizes](#).

Create a VM with [az vm create](#). The following example creates a VM named *myVM*:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image UbuntuLTS \
--size Standard_DS3_v2 \
--admin-username azureuser \
--generate-ssh-keys \
--nics myNic1 myNic2
```

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Add a NIC to a VM

The previous steps created a VM with multiple NICs. You can also add NICs to an existing VM with the Azure CLI. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly. If needed, you can [resize a VM](#).

Create another NIC with [az network nic create](#). The following example creates a NIC named *myNic3* connected to the back-end subnet and network security group created in the previous steps:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic3 \
--vnet-name myVnet \
--subnet mySubnetBackEnd \
--network-security-group myNetworkSecurityGroup
```

To add a NIC to an existing VM, first deallocate the VM with [az vm deallocate](#). The following example deallocates the VM named *myVM*:

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

Add the NIC with [az vm nic add](#). The following example adds *myNic3* to *myVM*:

```
az vm nic add \
--resource-group myResourceGroup \
--vm-name myVM \
--nics myNic3
```

Start the VM with [az vm start](#):

```
az vm start --resource-group myResourceGroup --name myVM
```

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Remove a NIC from a VM

To remove a NIC from an existing VM, first deallocate the VM with [az vm deallocate](#). The following example deallocates the VM named *myVM*.

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

Remove the NIC with [az vm nic remove](#). The following example removes *myNic3* from *myVM*.

```
az vm nic remove \
--resource-group myResourceGroup \
--vm-name myVM \
--nics myNic3
```

Start the VM with [az vm start](#):

```
az vm start --resource-group myResourceGroup --name myVM
```

## Create multiple NICs using Resource Manager templates

Azure Resource Manager templates use declarative JSON files to define your environment. You can read an [overview of Azure Resource Manager](#). Resource Manager templates provide a way to create multiple instances of a resource during deployment, such as creating multiple NICs. You use *copy* to specify the number of instances to create:

```
"copy": {
  "name": "multiplenics"
  "count": "[parameters('count')]"
}
```

Read more about [creating multiple instances using copy](#).

You can also use a `copyIndex()` to then append a number to a resource name, which allows you to create `myNic1`, `myNic2`, etc. The following shows an example of appending the index value:

```
"name": "[concat('myNic', copyIndex())]",
```

You can read a complete example of [creating multiple NICs using Resource Manager templates](#).

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Configure guest OS for multiple NICs

The previous steps created a virtual network and subnet, attached NICs, then created a VM. A public IP address and network security group rules that allow SSH traffic were not created. To configure the guest OS for multiple NICs, you need to allow remote connections and run commands locally on the VM.

To allow SSH traffic, create a network security group rule with [az network nsg rule create](#) as follows:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name allow_ssh \
--priority 101 \
--destination-port-ranges 22
```

Create a public IP address with [az network public-ip create](#) and assign it to the first NIC with [az network nic ip-config update](#):

```
az network public-ip create --resource-group myResourceGroup --name myPublicIP

az network nic ip-config update \
--resource-group myResourceGroup \
--nic-name myNic1 \
--name ipconfig1 \
--public-ip myPublicIP
```

To view the public IP address of the VM, use [az vm show](#) as follows::

```
az vm show --resource-group myResourceGroup --name myVM -d --query publicIps -o tsv
```

Now SSH to the public IP address of your VM. The default username provided in a previous step was *azureuser*. Provide your own username and public IP address:

```
ssh azureuser@137.117.58.232
```

To send to or from a secondary network interface, you have to manually add persistent routes to the operating system for each secondary network interface. In this article, *eth1* is the secondary interface. Instructions for adding persistent routes to the operating system vary by distro. See documentation for your distro for instructions.

When adding the route to the operating system, the gateway address is the first address of the subnet the network interface is in. For example, if the subnet has been assigned the range *10.0.2.0/24*, the gateway you specify for the route is *10.0.2.1* or if the subnet has been assigned the range *10.0.2.128/25*, the gateway you specify for the route is *10.0.2.129*. You can define a specific network for the route's destination, or specify a destination of *0.0.0.0*, if you want all traffic for the interface to go through the specified gateway. The gateway for each subnet is managed by the virtual network.

Once you've added the route for a secondary interface, verify that the route is in your route table with `route -n`. The following example output is for the route table that has the two network interfaces added to the VM in this article:

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	10.0.1.1	0.0.0.0	UG	0	0	0	eth0
0.0.0.0	10.0.2.1	0.0.0.0	UG	0	0	0	eth1
10.0.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
168.63.129.16	10.0.1.1	255.255.255.255	UGH	0	0	0	eth0
169.254.169.254	10.0.1.1	255.255.255.255	UGH	0	0	0	eth0

Confirm that the route you added persists across reboots by checking your route table again after a reboot. To test connectivity, you can enter the following command, for example, where *eth1* is the name of a secondary network interface:

```
ping bing.com -c 4 -I eth1
```

## Next steps

Review [Linux VM sizes](#) when trying to creating a VM with multiple NICs. Pay attention to the maximum number of NICs each VM size supports.

To further secure your VMs, use just in time VM access. This feature opens network security group rules for SSH traffic when needed, and for a defined period of time. For more information, see [Manage virtual machine access using just in time](#).

# Create a Windows VM with accelerated networking using Azure PowerShell

9/21/2022 • 8 minutes to read • [Edit Online](#)

## VM creation using the portal

Though this article provides steps to create a VM with accelerated networking using Azure PowerShell, you can also use the Azure portal to create a virtual machine that enables accelerated networking. When [creating a VM in the Azure Portal](#), in the **Create a virtual machine** page, choose the **Networking** tab. This tab has an option for **Accelerated networking**. If you have chosen a [supported operating system](#) and [VM size](#), this option is automatically set to **On**. Otherwise, the option is set to **Off**, and Azure displays the reason why it can't be enabled. You can also enable or disable accelerated networking through the portal after VM creation by navigating to the network interface and clicking the button at the top of the **Overview** blade.

### NOTE

Only supported operating systems can be enabled through the portal. If you are using a custom image, and your image supports accelerated networking, please create your VM using CLI or PowerShell.

After you create the VM, you can confirm whether accelerated networking is enabled. Follow these instructions:

1. Go to the [Azure portal](#) to manage your VMs. Search for and select **Virtual machines**.
2. In the virtual machine list, choose your new VM.
3. In the VM menu bar, choose **Networking**.

In the network interface information, next to the **Accelerated networking** label, the portal displays either **Disabled** or **Enabled** for the accelerated networking status.

## VM creation using PowerShell

Before you proceed, install [Azure PowerShell](#) version 1.0.0 or later. To find your currently installed version, run `Get-Module -ListAvailable Az`. If you need to install or upgrade, install the latest version of the Az module from the [PowerShell Gallery](#). In a PowerShell session, sign in to an Azure account using [Connect-AzAccount](#).

In the following examples, replace example parameter names with your own values. Example parameter names included *myResourceGroup*, *myNic*, and *myVM*.

### Create a virtual network

1. Create a resource group with [New-AzResourceGroup](#). The following command creates a resource group named *myResourceGroup* in the *centralus* location:

```
New-AzResourceGroup -Name "myResourceGroup" -Location "centralus"
```

2. Create a subnet configuration with [New-AzVirtualNetworkSubnetConfig](#). The following command creates a subnet named *mySubnet*.

```
$subnet = New-AzVirtualNetworkSubnetConfig `  
    -Name "mySubnet" `  
    -AddressPrefix "192.168.1.0/24"
```

3. Create a virtual network with [New-AzVirtualNetwork](#), with the *mySubnet* subnet.

```
$vnet = New-AzVirtualNetwork -ResourceGroupName "myResourceGroup" `  
    -Location "centralus" `  
    -Name "myVnet" `  
    -AddressPrefix "192.168.0.0/16" `  
    -Subnet $subnet
```

## Create a network security group

1. Create a network security group rule with [New-AzNetworkSecurityRuleConfig](#).

```
$rdp = New-AzNetworkSecurityRuleConfig `  
    -Name 'Allow-RDP-All' `  
    -Description 'Allow RDP' `  
    -Access Allow `  
    -Protocol Tcp `  
    -Direction Inbound `  
    -Priority 100 `  
    -SourceAddressPrefix * `  
    -SourcePortRange * `  
    -DestinationAddressPrefix * `  
    -DestinationPortRange 3389
```

2. Create a network security group with [New-AzNetworkSecurityGroup](#) and assign the *Allow-RDP-All* security rule to it. Aside from the *Allow-RDP-All* rule, the network security group contains several default rules. One default rule disables all inbound access from the internet. Once it's created, the *Allow-RDP-All* rule is assigned to the network security group so that you can remotely connect to the VM.

```
$nsg = New-AzNetworkSecurityGroup `  
    -ResourceGroupName myResourceGroup `  
    -Location centralus `  
    -Name "myNsg" `  
    -SecurityRules $rdp
```

3. Associate the network security group to the *mySubnet* subnet with [Set-AzVirtualNetworkSubnetConfig](#).

The rule in the network security group is effective for all resources deployed in the subnet.

```
Set-AzVirtualNetworkSubnetConfig `  
    -VirtualNetwork $vnet `  
    -Name 'mySubnet' `  
    -AddressPrefix "192.168.1.0/24" `  
    -NetworkSecurityGroup $nsg
```

## Create a network interface with accelerated networking

1. Create a public IP address with [New-AzPublicIpAddress](#). A public IP address is unnecessary if you don't plan to access the VM from the internet. However, it's required to complete the steps in this article.

```
$publicIp = New-AzPublicIpAddress  
    -ResourceGroupName myResourceGroup  
    -Name 'myPublicIp'  
    -location centralus  
    -AllocationMethod Dynamic
```

2. Create a network interface with [New-AzNetworkInterface](#) with accelerated networking enabled, and assign the public IP address to the network interface. The following example creates a network interface named *myNic* in the *mySubnet* subnet of the *myVnet* virtual network, assigning the *myPublicIp* public IP address to it:

```
$nic = New-AzNetworkInterface  
    -ResourceGroupName "myResourceGroup"  
    -Name "myNic"  
    -Location "centralus"  
    -SubnetId $vnet.Subnets[0].Id  
    -PublicIpAddressId $publicIp.Id  
    -EnableAcceleratedNetworking
```

### Create a VM and attach the network interface

1. Set your VM credentials to the `$cred` variable using [Get-Credential](#), which prompts you to sign in:

```
$cred = Get-Credential
```

2. Define your VM with [New-AzVMConfig](#). The following command defines a VM named *myVM* with a VM size that supports accelerated networking (*Standard\_DS4\_v2*):

```
$vmConfig = New-AzVMConfig -VMName "myVm" -VMSize "Standard_DS4_v2"
```

For a list of all VM sizes and characteristics, see [Windows VM sizes](#).

3. Create the rest of your VM configuration with [Set-AzVMOperatingSystem](#) and [Set-AzVMSourceImage](#). The following command creates a Windows Server 2016 VM:

```
$vmConfig = Set-AzVMOperatingSystem -VM $vmConfig  
    -Windows  
    -ComputerName "myVM"  
    -Credential $cred  
    -ProvisionVMAgent  
    -EnableAutoUpdate  
$vmConfig = Set-AzVMSourceImage -VM $vmConfig  
    -PublisherName "MicrosoftWindowsServer"  
    -Offer "WindowsServer"  
    -Skus "2016-Datacenter"  
    -Version "latest"
```

4. Attach the network interface that you previously created with [Add-AzVMNetworkInterface](#):

```
$vmConfig = Add-AzVMNetworkInterface -VM $vmConfig -Id $nic.Id
```

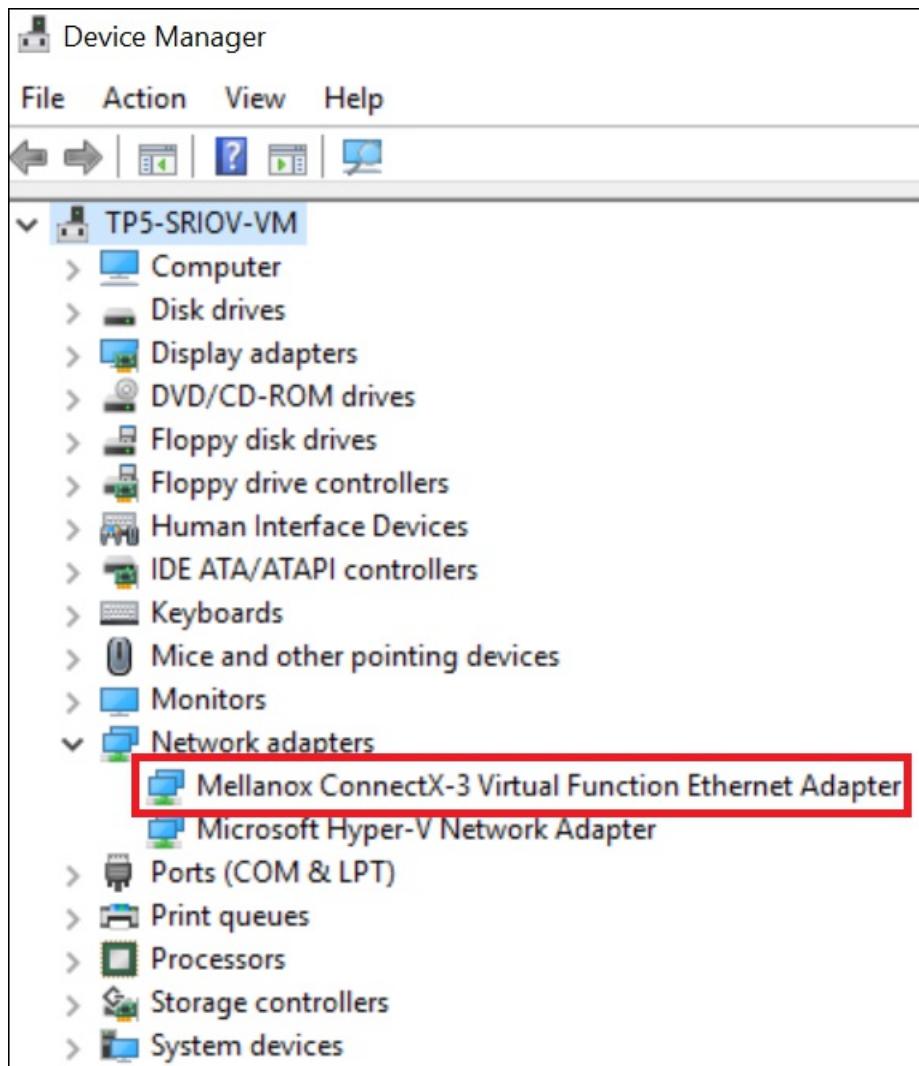
5. Create your VM with [New-AzVM](#).

```
New-AzVM -VM $vmConfig -ResourceGroupName "myResourceGroup" -Location "centralus"
```

## Confirm the Ethernet controller is installed in the Windows VM

Once you create the VM in Azure, connect to the VM and confirm that the Ethernet controller is installed in Windows.

1. Go to the [Azure portal](#) to manage your VMs. Search for and select **Virtual machines**.
2. In the virtual machine list, choose your new VM.
3. In the VM overview page, if the **Status** of the VM is listed as **Creating**, wait until Azure finishes creating the VM. The **Status** will be changed to **Running** after VM creation is complete.
4. From the VM overview toolbar, select **Connect > RDP > Download RDP File**.
5. Open the .rdp file, and then sign in to the VM with the credentials you entered in the [Create a VM and attach the network interface](#) section. If you've never connected to a Windows VM in Azure, see [Connect to virtual machine](#).
6. After the remote desktop session for your VM appears, right-click the Windows Start button and choose **Device Manager**.
7. In the **Device Manager** window, expand the **Network adapters** node.
8. Confirm that the **Mellanox ConnectX-3 Virtual Function Ethernet Adapter** appears, as shown in the following image:



Accelerated networking is now enabled for your VM.

#### NOTE

If the Mellanox adapter fails to start, open an administrator prompt in the remote desktop session and enter the following command:

```
netsh int tcp set global rss = enabled
```

## Enable accelerated networking on existing VMs

If you've created a VM without accelerated networking, you may enable this feature on an existing VM. The VM must support accelerated networking by meeting the following prerequisites, which are also outlined above:

- The VM must be a supported size for accelerated networking.
- The VM must be a supported Azure Gallery image (and kernel version for Linux).
- All VMs in an availability set or a virtual machine scale set must be stopped or deallocated before you enable accelerated networking on any NIC.

### Individual VMs and VMs in an availability set

1. Stop or deallocate the VM or, if an availability set, all the VMs in the set:

```
Stop-AzVM -ResourceGroup "myResourceGroup" -Name "myVM"
```

#### NOTE

When you create a VM individually, without an availability set, you only need to stop or deallocate the individual VM to enable accelerated networking. If your VM was created with an availability set, you must stop or deallocate all VMs contained in the availability set before enabling accelerated networking on any of the NICs, so that the VMs end up on a cluster that supports accelerated networking. The stop or deallocate requirement is unnecessary if you disable accelerated networking, because clusters that support accelerated networking also work fine with NICs that don't use accelerated networking.

2. Enable accelerated networking on the NIC of your VM:

```
$nic = Get-AzNetworkInterface -ResourceGroupName "myResourceGroup" `  
-Name "myNic"  
  
$nic.EnableAcceleratedNetworking = $true  
  
$nic | Set-AzNetworkInterface
```

3. Restart your VM or, if in an availability set, all the VMs in the set, and confirm that accelerated networking is enabled:

```
Start-AzVM -ResourceGroup "myResourceGroup" `  
-Name "myVM"
```

### Virtual machine scale set

A virtual machine scale set is slightly different, but it follows the same workflow.

1. Stop the VMs:

```
Stop-AzVmss -ResourceGroupName "myResourceGroup" `  
-VMScaleSetName "myScaleSet"
```

2. Update the accelerated networking property under the network interface:

```
$vmss = Get-AzVmss -ResourceGroupName "myResourceGroup" `  
-VMScaleSetName "myScaleSet"  
  
$vmss.VirtualMachineProfile.NetworkProfile.NetworkInterfaceConfigurations[0].EnableAcceleratedNetwork  
ing = $true  
  
Update-AzVmss -ResourceGroupName "myResourceGroup" `  
-VMScaleSetName "myScaleSet" `  
-VirtualMachineScaleSet $vmss
```

3. Set the applied updates to automatic so that the changes are immediately picked up:

```
$vmss.UpgradePolicy.Mode = "Automatic"  
  
Update-AzVmss -ResourceGroupName "myResourceGroup" `  
-VMScaleSetName "myScaleSet" `  
-VirtualMachineScaleSet $vmss
```

#### NOTE

A scale set has VM upgrades that apply updates using three different settings: automatic, rolling, and manual. In these instructions, the policy is set to automatic, so the scale set picks up the changes immediately after it restarts.

4. Restart the scale set:

```
Start-AzVmss -ResourceGroupName "myResourceGroup" `  
-VMScaleSetName "myScaleSet"
```

Once you restart, wait for the upgrades to finish. After the upgrades are done, the virtual function (VF) appears inside the VM. Make sure you're using a supported OS and VM size.

### Resizing existing VMs with accelerated networking

If a VM has accelerated networking enabled, you're only able to resize it to a VM that supports accelerated networking.

A VM with accelerated networking enabled can't be resized to a VM instance that doesn't support accelerated networking using the resize operation. Instead, to resize one of these VMs:

1. Stop or deallocate the VM. For an availability set or scale set, stop or deallocate all the VMs in the availability set or scale set.
2. Disable accelerated networking on the NIC of the VM. For an availability set or scale set, disable accelerated networking on the NICs of all VMs in the availability set or scale set.
3. After you disable accelerated networking, move the VM, availability set, or scale set to a new size that doesn't support accelerated networking, and then restart them.

## Next steps

- Learn [how Accelerated Networking works](#)
- Learn how to [create a VM with Accerelated Networking using Azure CLI](#)
- Improve latency with an [Azure proximity placement group](#)

# Create a Linux virtual machine with Accelerated Networking using Azure CLI

9/21/2022 • 8 minutes to read • [Edit Online](#)

## Portal creation

Though this article provides steps to create a virtual machine with accelerated networking using the Azure CLI, you can also [create a virtual machine with accelerated networking using the Azure portal](#). When creating a virtual machine in the portal, in the **Create a virtual machine** blade, choose the **Networking** tab. In this tab, there is an option for **Accelerated networking**. If you have chosen a [supported operating system](#) and [VM size](#), this option will automatically populate to "On." If not, it will populate the "Off" option for Accelerated Networking and give the user a reason why it isn't enabled.

You can also enable or disable accelerated networking through the portal after VM creation by navigating to the network interface and clicking the button at the top of the **Overview** blade.

### NOTE

The Accelerated Networking setting in the portal reflects the user-selected state. AccelNet allows choosing "Disabled" even if the VM size requires AccelNet. For those AccelNet-required VM sizes, AccelNet will be enabled at runtime regardless of the user setting seen in the portal.

Only supported operating systems can be enabled through the portal. If you're using a custom image, and your image supports Accelerated Networking, create your VM using CLI or PowerShell.

After the VM is created, you can confirm that Accelerated Networking is enabled by following the [confirmation instructions](#).

## CLI creation

### Create a virtual network

Install the latest [Azure CLI](#) and log in to an Azure account using [az login](#). In the following examples, replace example parameter names with your own values. Example parameter names included *myResourceGroup*, *myNic*, and *myVm*.

Create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroup* in the *centralus* location:

```
az group create --name myResourceGroup --location centralus
```

Select a supported Linux region listed in [Linux Accelerated Networking](#).

Create a virtual network with [az network vnet create](#). The following example creates a virtual network named *myVnet* with one subnet:

```
az network vnet create \
--resource-group myResourceGroup \
--name myVnet \
--address-prefix 192.168.0.0/16 \
--subnet-name mySubnet \
--subnet-prefix 192.168.1.0/24
```

## Create a network security group

Create a network security group with [az network nsg create](#). The following example creates a network security group named *myNetworkSecurityGroup*:

```
az network nsg create \
--resource-group myResourceGroup \
--name myNetworkSecurityGroup
```

The network security group contains several default rules, one of which disables all inbound access from the Internet. Open a port to allow SSH access to the virtual machine with [az network nsg rule create](#):

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name Allow-SSH-Internet \
--access Allow \
--protocol Tcp \
--direction Inbound \
--priority 100 \
--source-address-prefix Internet \
--source-port-range "*" \
--destination-address-prefix "*" \
--destination-port-range 22
```

## Create a network interface with Accelerated Networking

Create a public IP address with [az network public-ip create](#). A public IP address isn't required if you don't plan to access the VM from the Internet. However, it's required to complete the steps in this article.

```
az network public-ip create \
--name myPublicIp \
--resource-group myResourceGroup
```

Create a network interface with [az network nic create](#) with Accelerated Networking enabled. The following example creates a network interface named *myNic* in the *mySubnet* subnet of the *myVnet* virtual network and associates the *myNetworkSecurityGroup* network security group to the network interface:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic \
--vnet-name myVnet \
--subnet mySubnet \
--accelerated-networking true \
--public-ip-address myPublicIp \
--network-security-group myNetworkSecurityGroup
```

## Create a VM and attach the NIC

When you create the VM, specify the NIC you created with `--nics`. Select a size and distribution listed in [Linux accelerated networking](#).

Create a VM with [az vm create](#). The following example creates a VM named *myVM* with the UbuntuLTS image and a size that supports Accelerated Networking (*Standard\_DS4\_v2*):

```
az vm create \
    --resource-group myResourceGroup \
    --name myVM \
    --image UbuntuLTS \
    --size Standard_DS4_v2 \
    --admin-username azureuser \
    --generate-ssh-keys \
    --nics myNic
```

For a list of all VM sizes and characteristics, see [Linux VM sizes](#).

Once the VM is created, output similar to the following example output is returned. Take note of the **publicIpAddress**. This address is used to access the VM in subsequent steps.

```
{
  "fqdns": "",
  "id": "/subscriptions/<ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM",
  "location": "centralus",
  "macAddress": "00-0D-3A-23-9A-49",
  "powerState": "VM running",
  "privateIpAddress": "192.168.0.4",
  "publicIpAddress": "40.68.254.142",
  "resourceGroup": "myResourceGroup"
}
```

### Confirm that accelerated networking is enabled

Use the following command to create an SSH session with the VM. Replace `<your-public-ip-address>` with the public IP address assigned to the virtual machine that you created, and replace *azureuser* if you used a different value for `--admin-username` when you created the VM.

```
ssh azureuser@<your-public-ip-address>
```

From the Bash shell, enter `uname -r` and confirm that the kernel version is one of the following versions, or greater:

- **Ubuntu 16.04:** 4.11.0-1013
- **SLES SP3:** 4.4.92-6.18
- **RHEL:** 3.10.0-693, 2.6.32-573\*
- **CentOS:** 3.10.0-693

#### NOTE

Other kernel versions may be supported. For the most up to date list, reference the compatibility tables for each distribution at [Supported Linux and FreeBSD virtual machines for Hyper-V](#) and confirm that SR-IOV is supported.

Additional details can be found in the release notes for the [Linux Integration Services for Hyper-V and Azure](#). \* RHEL 6.7-6.10 are supported if the Mellanox VF version 4.5+ is installed before Linux Integration Services 4.3+.

Confirm that the Mellanox VF device is exposed to the VM with the `lspci` command. The returned output is similar to the following output:

```
0000:00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (AGP disabled) (rev 03)
0000:00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 01)
0000:00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
0000:00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 02)
0000:00:08.0 VGA compatible controller: Microsoft Corporation Hyper-V virtual VGA
0001:00:02.0 Ethernet controller: Mellanox Technologies MT27500/MT27520 Family [ConnectX-3/ConnectX-3 Pro
Virtual Function]
```

Check for activity on the VF (virtual function) with the `ethtool -S eth0 | grep vf_` command. If you receive output similar to the following sample output, accelerated networking is enabled and active.

```
vf_rx_packets: 992956
vf_rx_bytes: 2749784180
vf_tx_packets: 2656684
vf_tx_bytes: 1099443970
vf_tx_dropped: 0
```

Accelerated Networking is now enabled for your VM.

## Handle dynamic binding and revocation of virtual function

Applications must run over the synthetic NIC that is exposed in VM. If the application runs directly over the VF NIC, it doesn't receive **all** packets that are destined to the VM, since some packets show up over the synthetic interface. If you run an application over the synthetic NIC, it guarantees that the application receives **all** packets that are destined to it. It also makes sure that the application keeps running, even if the VF is revoked during host servicing. Applications binding to the synthetic NIC is a **mandatory** requirement for all applications taking advantage of **Accelerated Networking**.

For more details on application binding requirements, see [How Accelerated Networking works in Linux and FreeBSD VMs](#).

## Enable Accelerated Networking on existing VMs

If you've created a VM without Accelerated Networking, it's possible to enable this feature on an existing VM. The VM must support Accelerated Networking by meeting the following prerequisites that are also outlined:

- The VM must be a supported size for Accelerated Networking
- The VM must be a supported Azure Gallery image (and kernel version for Linux)
- All VMs in an availability set or VMSS must be stopped/deallocated before enabling Accelerated Networking on any NIC

### Individual VMs & VMs in an availability set

First stop/deallocate the VM or, if an Availability Set, all the VMs in the Set:

```
az vm deallocate \
--resource-group myResourceGroup \
--name myVM
```

If your VM was created individually without an availability set, you only must stop or deallocate the individual VM to enable Accelerated Networking. If your VM was created with an availability set, all VMs contained in the set must be stopped or deallocated before enabling Accelerated Networking on any of the NICs.

Once stopped, enable Accelerated Networking on the NIC of your VM:

```
az network nic update \
--name myNic \
--resource-group myResourceGroup \
--accelerated-networking true
```

Restart your VM or, if in an Availability Set, all the VMs in the Set and confirm that Accelerated Networking is enabled:

```
az vm start --resource-group myResourceGroup \
--name myVM
```

## VMSS

VMSS is slightly different but follows the same workflow. First, stop the VMs:

```
az vmss deallocate \
--name myvmss \
--resource-group myrg
```

Once the VMs are stopped, update the Accelerated Networking property under the network interface:

```
az vmss update --name myvmss \
--resource-group myrg \
--set
virtualMachineProfile.networkProfile.networkInterfaceConfigurations[0].enableAcceleratedNetworking=true
```

### NOTE

A VMSS has VM upgrades that apply updates using three different settings, automatic, rolling, and manual. In these instructions, the policy is set to automatic so that the VMSS will pick up the changes immediately after reboot. To set it to automatic so that the changes are immediately picked up:

```
az vmss update \
--name myvmss \
--resource-group myrg \
--set upgradePolicy.mode="automatic"
```

Finally, restart the VMSS:

```
az vmss start \
--name myvmss \
--resource-group myrg
```

Once you restart, wait for the upgrades to finish but once completed, the VF appears inside the VM. (Make sure you're using a supported OS and VM size.)

## Resizing existing VMs with Accelerated Networking

VMs with Accelerated Networking enabled can only be resized to VMs that support Accelerated Networking.

A VM with Accelerated Networking enabled can't be resized to a VM instance that doesn't support Accelerated Networking using the resize operation. Instead, to resize one of these VMs:

- Stop/Deallocate the VM or if in an availability set/VMSS, stop/deallocate all the VMs in the set/VMSS.

- Accelerated Networking must be disabled on the NIC of the VM or if in an availability set/VMSS, all VMs in the set/VMSS.
- Once Accelerated Networking is disabled, the VM/availability set/VMSS can be moved to a new size that doesn't support Accelerated Networking and restarted.

## Next steps

- Learn [how Accelerated Networking works](#)
- Learn how to [create a VM with Accelerated Networking in PowerShell](#)
- Improve latency with an [Azure proximity placement group](#)

# Set up DPDK in a Linux virtual machine

9/21/2022 • 7 minutes to read • [Edit Online](#)

Data Plane Development Kit (DPDK) on Azure offers a faster user-space packet processing framework for performance-intensive applications. This framework bypasses the virtual machine's kernel network stack.

In typical packet processing that uses the kernel network stack, the process is interrupt-driven. When the network interface receives incoming packets, there is a kernel interrupt to process the packet and a context switch from the kernel space to the user space. DPDK eliminates context switching and the interrupt-driven method in favor of a user-space implementation that uses poll mode drivers for fast packet processing.

DPDK consists of sets of user-space libraries that provide access to lower-level resources. These resources can include hardware, logical cores, memory management, and poll mode drivers for network interface cards.

DPDK can run on Azure virtual machines that are supporting multiple operating system distributions. DPDK provides key performance differentiation in driving network function virtualization implementations. These implementations can take the form of network virtual appliances (NVAs), such as virtual routers, firewalls, VPNs, load balancers, evolved packet cores, and denial-of-service (DDoS) applications.

## Benefit

**Higher packets per second (PPS):** Bypassing the kernel and taking control of packets in the user space reduces the cycle count by eliminating context switches. It also improves the rate of packets that are processed per second in Azure Linux virtual machines.

## Supported operating systems minimum versions

The following distributions from the Azure Marketplace are supported:

LINUX OS	KERNEL VERSION
Ubuntu 18.04	4.15.0-1014-azure+
SLES 15 SP1	4.12.14-8.19-azure+
RHEL 7.5	3.10.0-862.11.6.el7.x86_64+
CentOS 7.5	3.10.0-862.11.6.el7.x86_64+
Debian 10	4.19.0-1-cloud+

The noted versions are the minimum requirements. Newer versions are supported too.

## Custom kernel support

For any Linux kernel version that's not listed, see [Patches for building an Azure-tuned Linux kernel](#). For more information, you can also contact [aznetdpdk@microsoft.com](mailto:aznetdpdk@microsoft.com).

## Region support

All Azure regions support DPDK.

## Prerequisites

Accelerated networking must be enabled on a Linux virtual machine. The virtual machine should have at least two network interfaces, with one interface for management. Enabling Accelerated networking on management interface is not recommended. Learn how to [create a Linux virtual machine with accelerated networking enabled](#).

On virtual machines that are using InfiniBand, ensure the appropriate `mlx4_ib` or `mlx5_ib` drivers are loaded, see [Enable InfiniBand](#).

## Install DPDK via system package (recommended)

### Ubuntu 18.04

```
sudo add-apt-repository ppa:canonical-server/server-backports -y  
sudo apt-get update  
sudo apt-get install -y dpdk
```

### Ubuntu 20.04 and newer

```
sudo apt-get install -y dpdk
```

### Debian 10 and newer

```
sudo apt-get install -y dpdk
```

## Install DPDK manually (not recommended)

### Install build dependencies

#### Ubuntu 18.04

```
sudo add-apt-repository ppa:canonical-server/server-backports -y  
sudo apt-get update  
sudo apt-get install -y build-essential librdmacm-dev libnuma-dev libmnl-dev meson
```

#### Ubuntu 20.04 and newer

```
sudo apt-get install -y build-essential librdmacm-dev libnuma-dev libmnl-dev meson
```

#### Debian 10 and newer

```
sudo apt-get install -y build-essential librdmacm-dev libnuma-dev libmnl-dev meson
```

#### RHEL7.5/CentOS 7.5

```
yum -y groupinstall "Infiniband Support"  
sudo dracut --add-drivers "mlx4_en mlx4_ib mlx5_ib" -f  
yum install -y gcc kernel-devel-`uname -r` numactl-devel.x86_64 librdmacm-devel libmnl-devel meson
```

#### SLES 15 SP1

#### Azure kernel

```
zypper \
--no-gpg-checks \
--non-interactive \
--gpg-auto-import-keys install kernel-azure kernel-devel-azure gcc make libnuma-devel numactl librdmacm1
rdma-core-devel meson
```

## Default kernel

```
zypper \
--no-gpg-checks \
--non-interactive \
--gpg-auto-import-keys install kernel-default-devel gcc make libnuma-devel numactl librdmacm1 rdma-core-
devel meson
```

## Compile and install DPDK manually

1. [Download the latest DPDK](#). Version 19.11 LTS or newer is required for Azure.
2. Build the default config with `meson builddir`.
3. Compile with `ninja -C builddir`.
4. Install with `DESTDIR=<output folder> ninja -C builddir install`.

## Configure the runtime environment

After restarting, run the following commands once:

1. Hugepages
  - Configure hugepage by running the following command, once for each numa node:

```
echo 1024 | sudo tee /sys/devices/system/node/node*/hugepages/hugepages-2048kB/nr_hugepages
```
  - Create a directory for mounting with `mkdir /mnt/huge`.
  - Mount hugepages with `mount -t hugetlbfs nodev /mnt/huge`.
  - Check that hugepages are reserved with `grep Huge /proc/meminfo`.

[NOTE] There is a way to modify the grub file so that hugepages are reserved on boot by following the [instructions](#) for the DPDK. The instructions are at the bottom of the page. When you're using an Azure Linux virtual machine, modify files under `/etc/config/grub.d` instead, to reserve hugepages across reboots.
2. MAC & IP addresses: Use `ifconfig -a` to view the MAC and IP address of the network interfaces. The *VF* network interface and *NETVSC* network interface have the same MAC address, but only the *NETVSC* network interface has an IP address. *VF* interfaces are running as subordinate interfaces of *NETVSC* interfaces.
3. PCI addresses
  - Use `ethtool -i <vf interface name>` to find out which PCI address to use for *VF*.
  - If *eth0* has accelerated networking enabled, make sure that testpmd doesn't accidentally take over the *VF* pci device for *eth0*. If the DPDK application accidentally takes over the management network interface and causes you to lose your SSH connection, use the serial console to stop the DPDK application. You can also use the serial console to stop or start the virtual machine.
4. Load *ibuverbs* on each reboot with `modprobe -a ib_uverbs`. For SLES 15 only, also load *mlx4\_ib* with

```
modprobe -a mlx4_ib .
```

## Failsafe PMD

DPDK applications must run over the failsafe PMD that is exposed in Azure. If the application runs directly over the VFPM, it doesn't receive all packets that are destined to the VM, since some packets show up over the synthetic interface.

If you run a DPDK application over the failsafe PMD, it guarantees that the application receives all packets that are destined to it. It also makes sure that the application keeps running in DPDK mode, even if the VF is revoked when the host is being serviced. For more information about failsafe PMD, see [Fail-safe poll mode driver library](#).

## Run testpmd

To run testpmd in root mode, use `sudo` before the `testpmd` command.

### Basic: Sanity check, failsafe adapter initialization

1. Run the following commands to start a single port testpmd application:

```
testpmd -w <pci address from previous step> \
--vdev="net_vdev_netvsc0,iface=eth1" \
-- -i \
--port-topology=chained
```

2. Run the following commands to start a dual port testpmd application:

```
testpmd -w <pci address nic1> \
-w <pci address nic2> \
--vdev="net_vdev_netvsc0,iface=eth1" \
--vdev="net_vdev_netvsc1,iface=eth2" \
-- -i
```

If you're running testpmd with more than two NICs, the `--vdev` argument follows this pattern:

```
net_vdev_netvsc<id>,iface=<vf's pairing eth> .
```

3. After it's started, run `show port info all` to check port information. You should see one or two DPDK ports that are `net_failsafe` (not `net_mlx4`).
4. Use `start <port> /stop <port>` to start traffic.

The previous commands start `testpmd` in interactive mode, which is recommended for trying out testpmd commands.

### Basic: Single sender/single receiver

The following commands periodically print the packets per second statistics:

1. On the TX side, run the following command:

```
testpmd \
-l <core-list> \
-n <num of mem channels> \
-w <pci address of the device you plan to use> \
--vdev="net_vdev_netvsc<id>,iface=<the iface to attach to>" \
--port-topology=chained \
--nb-cores <number of cores to use for test pmd> \
--forward-mode=txonly \
--eth-peer=<port id>,<receiver peer MAC address> \
--stats-period <display interval in seconds>
```

2. On the RX side, run the following command:

```
testpmd \
-l <core-list> \
-n <num of mem channels> \
-w <pci address of the device you plan to use> \
--vdev="net_vdev_netvsc<id>,iface=<the iface to attach to>" \
--port-topology=chained \
--nb-cores <number of cores to use for test pmd> \
--forward-mode=rxonly \
--eth-peer=<port id>,<sender peer MAC address> \
--stats-period <display interval in seconds>
```

When you're running the previous commands on a virtual machine, change *IP\_SRC\_ADDR* and *IP\_DST\_ADDR* in `app/test-pmd/txonly.c` to match the actual IP address of the virtual machines before you compile. Otherwise, the packets are dropped before reaching the receiver.

### **Advanced: Single sender/single forwarder**

The following commands periodically print the packets per second statistics:

1. On the TX side, run the following command:

```
testpmd \
-l <core-list> \
-n <num of mem channels> \
-w <pci address of the device you plan to use> \
--vdev="net_vdev_netvsc<id>,iface=<the iface to attach to>" \
--port-topology=chained \
--nb-cores <number of cores to use for test pmd> \
--forward-mode=txonly \
--eth-peer=<port id>,<receiver peer MAC address> \
--stats-period <display interval in seconds>
```

2. On the FWD side, run the following command:

```
testpmd \
-l <core-list> \
-n <num of mem channels> \
-w <pci address NIC1> \
-w <pci address NIC2> \
--vdev="net_vdev_netvsc<id>,iface=<the iface to attach to>" \
--vdev="net_vdev_netvsc<2nd id>,iface=<2nd iface to attach to>" (you need as many --vdev arguments as the number of devices used by testpmd, in this case) \
--nb-cores <number of cores to use for test pmd> \
--forward-mode=io \
--eth-peer=<recv port id>,<sender peer MAC address> \
--stats-period <display interval in seconds>
```

When you're running the previous commands on a virtual machine, change *IP\_SRC\_ADDR* and *IP\_DST\_ADDR* in `app/test-pmd/txonly.c` to match the actual IP address of the virtual machines before you compile. Otherwise, the packets are dropped before reaching the forwarder. You won't be able to have a third machine receive forwarded traffic, because the *testpmd* forwarder doesn't modify the layer-3 addresses, unless you make some code changes.

## References

- [EAL options](#)
- [Testpmd commands](#)
- [Packet dump commands](#)

# TCP/IP performance tuning for Azure VMs

9/21/2022 • 23 minutes to read • [Edit Online](#)

This article discusses common TCP/IP performance tuning techniques and some things to consider when you use them for virtual machines running on Azure. It will provide a basic overview of the techniques and explore how they can be tuned.

## Common TCP/IP tuning techniques

### MTU, fragmentation, and large send offload

#### MTU

The maximum transmission unit (MTU) is the largest size frame (packet), specified in bytes, that can be sent over a network interface. The MTU is a configurable setting. The default MTU used on Azure VMs, and the default setting on most network devices globally, is 1,500 bytes.

#### Fragmentation

Fragmentation occurs when a packet is sent that exceeds the MTU of a network interface. The TCP/IP stack will break the packet into smaller pieces (fragments) that conform to the interface's MTU. Fragmentation occurs at the IP layer and is independent of the underlying protocol (such as TCP). When a 2,000-byte packet is sent over a network interface with an MTU of 1,500, the packet will be broken down into one 1,500-byte packet and one 500-byte packet.

Network devices in the path between a source and destination can either drop packets that exceed the MTU or fragment the packet into smaller pieces.

#### The Don't Fragment bit in an IP packet

The Don't Fragment (DF) bit is a flag in the IP protocol header. The DF bit indicates that network devices on the path between the sender and receiver must not fragment the packet. This bit could be set for many reasons. (See the "Path MTU Discovery" section of this article for one example.) When a network device receives a packet with the Don't Fragment bit set, and that packet exceeds the device's interface MTU, the standard behavior is for the device to drop the packet. The device sends an ICMP Fragmentation Needed message back to the original source of the packet.

#### Performance implications of fragmentation

Fragmentation can have negative performance implications. One of the main reasons for the effect on performance is the CPU/memory impact of the fragmentation and reassembly of packets. When a network device needs to fragment a packet, it will have to allocate CPU/memory resources to perform fragmentation.

The same thing happens when the packet is reassembled. The network device has to store all the fragments until they're received so it can reassemble them into the original packet. This process of fragmentation and reassembly can also cause latency.

The other possible negative performance implication of fragmentation is that fragmented packets might arrive out of order. When packets are received out of order, some types of network devices can drop them. When that happens, the whole packet has to be retransmitted.

Fragments are typically dropped by security devices like network firewalls or when a network device's receive buffers are exhausted. When a network device's receive buffers are exhausted, a network device is attempting to reassemble a fragmented packet but doesn't have the resources to store and reassemble the packet.

Fragmentation can be seen as a negative operation, but support for fragmentation is necessary when you're connecting diverse networks over the internet.

## **Benefits and consequences of modifying the MTU**

Generally speaking, you can create a more efficient network by increasing the MTU. Every packet that's transmitted has header information that's added to the original packet. When fragmentation creates more packets, there's more header overhead, and that makes the network less efficient.

Here's an example. The Ethernet header size is 14 bytes plus a 4-byte frame check sequence to ensure frame consistency. If one 2,000-byte packet is sent, 18 bytes of Ethernet overhead is added on the network. If the packet is fragmented into a 1,500-byte packet and a 500-byte packet, each packet will have 18 bytes of Ethernet header, a total of 36 bytes.

Keep in mind that increasing the MTU won't necessarily create a more efficient network. If an application sends only 500-byte packets, the same header overhead will exist whether the MTU is 1,500 bytes or 9,000 bytes. The network will become more efficient only if it uses larger packet sizes that are affected by the MTU.

### **Azure and VM MTU**

The default MTU for Azure VMs is 1,500 bytes. The Azure Virtual Network stack will attempt to fragment a packet at 1,400 bytes.

Note that the Virtual Network stack isn't inherently inefficient because it fragments packets at 1,400 bytes even though VMs have an MTU of 1,500. A large percentage of network packets are much smaller than 1,400 or 1,500 bytes.

### **Azure and fragmentation**

Virtual Network stack is set up to drop "out of order fragments," that is, fragmented packets that don't arrive in their original fragmented order. These packets are dropped mainly because of a network security vulnerability announced in November 2018 called FragmentSmack.

FragmentSmack is a defect in the way the Linux kernel handled reassembly of fragmented IPv4 and IPv6 packets. A remote attacker could use this flaw to trigger expensive fragment reassembly operations, which could lead to increased CPU and a denial of service on the target system.

### **Tune the MTU**

You can configure an Azure VM MTU, as you can in any other operating system. But you should consider the fragmentation that occurs in Azure, described above, when you're configuring an MTU.

We don't encourage customers to increase VM MTUs. This discussion is meant to explain the details of how Azure implements MTU and performs fragmentation.

#### **IMPORTANT**

Increasing MTU isn't known to improve performance and could have a negative effect on application performance.

### **Large send offload**

Large send offload (LSO) can improve network performance by offloading the segmentation of packets to the Ethernet adapter. When LSO is enabled, the TCP/IP stack creates a large TCP packet and sends it to the Ethernet adapter for segmentation before forwarding it. The benefit of LSO is that it can free the CPU from segmenting packets into sizes that conform to the MTU and offload that processing to the Ethernet interface where it's performed in hardware. To learn more about the benefits of LSO, see [Supporting large send offload](#).

When LSO is enabled, Azure customers might see large frame sizes when they perform packet captures. These large frame sizes might lead some customers to think fragmentation is occurring or that a large MTU is being used when it's not. With LSO, the Ethernet adapter can advertise a larger maximum segment size (MSS) to the TCP/IP stack to create a larger TCP packet. This entire non-segmented frame is then forwarded to the Ethernet adapter and would be visible in a packet capture performed on the VM. But the packet will be broken down into many smaller frames by the Ethernet adapter, according to the Ethernet adapter's MTU.

### **TCP MSS window scaling and PMTUD**

## TCP maximum segment size

TCP maximum segment size (MSS) is a setting that limits the size of TCP segments, which avoids fragmentation of TCP packets. Operating systems will typically use this formula to set MSS:

$$\text{MSS} = \text{MTU} - (\text{IP header size} + \text{TCP header size})$$

The IP header and the TCP header are 20 bytes each, or 40 bytes total. So an interface with an MTU of 1,500 will have an MSS of 1,460. But the MSS is configurable.

This setting is agreed to in the TCP three-way handshake when a TCP session is set up between a source and a destination. Both sides send an MSS value, and the lower of the two is used for the TCP connection.

Keep in mind that the MTUs of the source and destination aren't the only factors that determine the MSS value. Intermediary network devices, like VPN gateways, including Azure VPN Gateway, can adjust the MTU independently of the source and destination to ensure optimal network performance.

### Path MTU Discovery

MSS is negotiated, but it might not indicate the actual MSS that can be used. This is because other network devices in the path between the source and the destination might have a lower MTU value than the source and destination. In this case, the device whose MTU is smaller than the packet will drop the packet. The device will send back an ICMP Fragmentation Needed (Type 3, Code 4) message that contains its MTU. This ICMP message allows the source host to reduce its Path MTU appropriately. The process is called Path MTU Discovery (PMTUD).

The PMTUD process is inefficient and affects network performance. When packets are sent that exceed a network path's MTU, the packets need to be retransmitted with a lower MSS. If the sender doesn't receive the ICMP Fragmentation Needed message, maybe because of a network firewall in the path (commonly referred to as a *PMTUD blackhole*), the sender doesn't know it needs to lower the MSS and will continuously retransmit the packet. This is why we don't recommend increasing the Azure VM MTU.

### VPN and MTU

If you use VMs that perform encapsulation (like IPsec VPNs), there are some additional considerations regarding packet size and MTU. VPNs add more headers to packets, which increases the packet size and requires a smaller MSS.

For Azure, we recommend that you set TCP MSS clamping to 1,350 bytes and tunnel interface MTU to 1,400. For more information, see the [VPN devices and IPsec/IKE parameters page](#).

## Latency, round-trip time, and TCP window scaling

### Latency and round-trip time

Network latency is governed by the speed of light over a fiber optic network. Network throughput of TCP is also effectively governed by the round-trip time (RTT) between two network devices.

ROUTE	DISTANCE	ONE-WAY TIME	RTT
New York to San Francisco	4,148 km	21 ms	42 ms
New York to London	5,585 km	28 ms	56 ms
New York to Sydney	15,993 km	80 ms	160 ms

This table shows the straight-line distance between two locations. In networks, the distance is typically longer than the straight-line distance. Here's a simple formula to calculate minimum RTT as governed by the speed of light:

$$\text{minimum RTT} = 2 * (\text{Distance in kilometers} / \text{Speed of propagation})$$

You can use 200 for the speed of propagation. This is the distance, in kilometers, that light travels in 1

millisecond.

Let's take New York to San Francisco as an example. The straight-line distance is 4,148 km. Plugging that value into the equation, we get the following:

$$\text{Minimum RTT} = 2 * (4,148 / 200)$$

The output of the equation is in milliseconds.

If you want to get the best network performance, the logical option is to select destinations with the shortest distance between them. You should also design your virtual network to optimize the path of traffic and reduce latency. For more information, see the "Network design considerations" section of this article.

#### Latency and round-trip time effects on TCP

Round-trip time has a direct effect on maximum TCP throughput. In TCP protocol, *window size* is the maximum amount of traffic that can be sent over a TCP connection before the sender needs to receive acknowledgement from the receiver. If the TCP MSS is set to 1,460 and the TCP window size is set to 65,535, the sender can send 45 packets before it has to receive acknowledgement from the receiver. If the sender doesn't get acknowledgement, it will retransmit the data. Here's the formula:

$$\text{TCP window size} / \text{TCP MSS} = \text{packets sent}$$

In this example, 65,535 / 1,460 is rounded up to 45.

This "waiting for acknowledgement" state, a mechanism to ensure reliable delivery of data, is what causes RTT to affect TCP throughput. The longer the sender waits for acknowledgement, the longer it needs to wait before sending more data.

Here's the formula for calculating the maximum throughput of a single TCP connection:

$$\text{Window size} / (\text{RTT latency in milliseconds} / 1,000) = \text{maximum bytes/second}$$

This table shows the maximum megabytes/second throughput of a single TCP connection. (For readability, megabytes is used for the unit of measure.)

TCP WINDOW SIZE (BYTES)	RTT LATENCY (MS)	MAXIMUM MEGABYTE/SECOND THROUGHPUT	MAXIMUM MEGABIT/SECOND THROUGHPUT
65,535	1	65.54	524.29
65,535	30	2.18	17.48
65,535	60	1.09	8.74
65,535	90	.73	5.83
65,535	120	.55	4.37

If packets are lost, the maximum throughput of a TCP connection will be reduced while the sender retransmits data it has already sent.

#### TCP window scaling

TCP window scaling is a technique that dynamically increases the TCP window size to allow more data to be sent before an acknowledgement is required. In the previous example, 45 packets would be sent before an acknowledgement was required. If you increase the number of packets that can be sent before an acknowledgement is needed, you're reducing the number of times a sender is waiting for acknowledgement, which increases the TCP maximum throughput.

This table illustrates those relationships:

TCP WINDOW SIZE (BYTES)	RTT LATENCY (MS)	MAXIMUM MEGABYTE/SECOND THROUGHPUT	MAXIMUM MEGABIT/SECOND THROUGHPUT
65,535	30	2.18	17.48
131,070	30	4.37	34.95
262,140	30	8.74	69.91
524,280	30	17.48	139.81

But the TCP header value for TCP window size is only 2 bytes long, which means the maximum value for a receive window is 65,535. To increase the maximum window size, a TCP window scale factor was introduced.

The scale factor is also a setting that you can configure in an operating system. Here's the formula for calculating the TCP window size by using scale factors:

```
TCP window size = TCP window size in bytes \* (2^scale factor)
```

Here's the calculation for a window scale factor of 3 and a window size of 65,535:

```
65,535 \* (2^3) = 262,140 bytes
```

A scale factor of 14 results in a TCP window size of 14 (the maximum offset allowed). The TCP window size will be 1,073,725,440 bytes (8.5 gigabits).

#### Support for TCP window scaling

Windows can set different scaling factors for different connection types. (Classes of connections include datacenter, internet, and so on.) You use the `Get-NetTCPConnection` PowerShell command to view the window scaling connection type:

```
Get-NetTCPConnection
```

You can use the `Get-NetTCPSetting` PowerShell command to view the values of each class:

```
Get-NetTCPSetting
```

You can set the initial TCP window size and TCP scaling factor in Windows by using the `Set-NetTCPSetting` PowerShell command. For more information, see [Set-NetTCPSetting](#).

```
Set-NetTCPSetting
```

These are the effective TCP settings for `AutoTuningLevel`:

AUTOTUNINGLEVEL	SCALING FACTOR	SCALING MULTIPLIER	FORMULA TO CALCULATE MAXIMUM WINDOW SIZE
Disabled	None	None	Window size
Restricted	4	$2^4$	Window size * $(2^4)$

AUTOTUNINGLEVEL	SCALING FACTOR	SCALING MULTIPLIER	FORMULA TO CALCULATE MAXIMUM WINDOW SIZE
Highly restricted	2	$2^2$	Window size * $(2^2)$
Normal	8	$2^8$	Window size * $(2^8)$
Experimental	14	$2^{14}$	Window size * $(2^{14})$

These settings are the most likely to affect TCP performance, but keep in mind that many other factors across the internet, outside the control of Azure, can also affect TCP performance.

#### Increase MTU size

Because a larger MTU means a larger MSS, you might wonder whether increasing the MTU can increase TCP performance. Probably not. There are pros and cons to packet size beyond just TCP traffic. As discussed earlier, the most important factors affecting TCP throughput performance are TCP window size, packet loss, and RTT.

#### IMPORTANT

We don't recommend that Azure customers change the default MTU value on virtual machines.

## Accelerated networking and receive side scaling

### Accelerated networking

Virtual machine network functions have historically been CPU intensive on both the guest VM and the hypervisor/host. Every packet that transits through the host is processed in software by the host CPU, including all virtual network encapsulation and decapsulation. So the more traffic that goes through the host, the higher the CPU load. And if the host CPU is busy with other operations, that will also affect network throughput and latency. Azure addresses this issue with accelerated networking.

Accelerated networking provides consistent ultralow network latency via the in-house programmable hardware of Azure and technologies like SR-IOV. Accelerated networking moves much of the Azure software-defined networking stack off the CPUs and into FPGA-based SmartNICs. This change enables end-user applications to reclaim compute cycles, which puts less load on the VM, decreasing jitter and inconsistency in latency. In other words, performance can be more deterministic.

Accelerated networking improves performance by allowing the guest VM to bypass the host and establish a datapath directly with a host's SmartNIC. Here are some benefits of accelerated networking:

- **Lower latency / higher packets per second (pps):** Removing the virtual switch from the datapath eliminates the time packets spend in the host for policy processing and increases the number of packets that can be processed in the VM.
- **Reduced jitter:** Virtual switch processing depends on the amount of policy that needs to be applied and the workload of the CPU that's doing the processing. Offloading the policy enforcement to the hardware removes that variability by delivering packets directly to the VM, eliminating the host-to-VM communication and all software interrupts and context switches.
- **Decreased CPU utilization:** Bypassing the virtual switch in the host leads to less CPU utilization for processing network traffic.

To use accelerated networking, you need to explicitly enable it on each applicable VM. See [Create a Linux virtual machine with Accelerated Networking](#) for instructions.

### Receive side scaling

Receive side scaling (RSS) is a network driver technology that distributes the receiving of network traffic more

efficiently by distributing receive processing across multiple CPUs in a multiprocessor system. In simple terms, RSS allows a system to process more received traffic because it uses all available CPUs instead of just one. For a more technical discussion of RSS, see [Introduction to receive side scaling](#).

To get the best performance when accelerated networking is enabled on a VM, you need to enable RSS. RSS can also provide benefits on VMs that don't use accelerated networking. For an overview of how to determine if RSS is enabled and how to enable it, see [Optimize network throughput for Azure virtual machines](#).

### TCP TIME\_WAIT and TIME\_WAIT assassination

TCP TIME\_WAIT is another common setting that affects network and application performance. On busy VMs that are opening and closing many sockets, either as clients or as servers (Source IP:Source Port + Destination IP:Destination Port), during the normal operation of TCP, a given socket can end up in a TIME\_WAIT state for a long time. The TIME\_WAIT state is meant to allow any additional data to be delivered on a socket before closing it. So TCP/IP stacks generally prevent the reuse of a socket by silently dropping the client's TCP SYN packet.

The amount of time a socket is in TIME\_WAIT is configurable. It could range from 30 seconds to 240 seconds. Sockets are a finite resource, and the number of sockets that can be used at any given time is configurable. (The number of available sockets is typically about 30,000.) If the available sockets are consumed, or if clients and servers have mismatched TIME\_WAIT settings, and a VM tries to reuse a socket in a TIME\_WAIT state, new connections will fail as TCP SYN packets are silently dropped.

The value for port range for outbound sockets is usually configurable within the TCP/IP stack of an operating system. The same thing is true for TCP TIME\_WAIT settings and socket reuse. Changing these numbers can potentially improve scalability. But, depending on the situation, these changes could cause interoperability issues. You should be careful if you change these values.

You can use TIME\_WAIT assassination to address this scaling limitation. TIME\_WAIT assassination allows a socket to be reused in certain situations, like when the sequence number in the IP packet of the new connection exceeds the sequence number of the last packet from the previous connection. In this case, the operating system will allow the new connection to be established (it will accept the new SYN/ACK) and force close the previous connection that was in a TIME\_WAIT state. This capability is supported on Windows VMs in Azure. To learn about support in other VMs, check with the OS vendor.

To learn about configuring TCP TIME\_WAIT settings and source port range, see [Settings that can be modified to improve network performance](#).

## Virtual network factors that can affect performance

### VM maximum outbound throughput

Azure provides a variety of VM sizes and types, each with a different mix of performance capabilities. One of these capabilities is network throughput (or bandwidth), which is measured in megabits per second (Mbps). Because virtual machines are hosted on shared hardware, the network capacity needs to be shared fairly among the virtual machines using the same hardware. Larger virtual machines are allocated more bandwidth than smaller virtual machines.

The network bandwidth allocated to each virtual machine is metered on egress (outbound) traffic from the virtual machine. All network traffic leaving the virtual machine is counted toward the allocated limit, regardless of destination. For example, if a virtual machine has a 1,000-Mbps limit, that limit applies whether the outbound traffic is destined for another virtual machine in the same virtual network or one outside of Azure.

Ingress is not metered or limited directly. But there are other factors, like CPU and storage limits, that can affect a virtual machine's ability to process incoming data.

Accelerated networking is designed to improve network performance, including latency, throughput, and CPU utilization. Accelerated networking can improve a virtual machine's throughput, but it can do that only up to the virtual machine's allocated bandwidth.

Azure virtual machines have at least one network interface attached to them. They might have several. The bandwidth allocated to a virtual machine is the sum of all outbound traffic across all network interfaces attached to the machine. In other words, the bandwidth is allocated on a per-virtual machine basis, regardless of how many network interfaces are attached to the machine.

Expected outbound throughput and the number of network interfaces supported by each VM size are detailed in [Sizes for Windows virtual machines in Azure](#). To see maximum throughput, select a type, like **General purpose**, and then find the section about the size series on the resulting page (for example, "Dv2-series"). For each series, there's a table that provides networking specifications in the last column, which is titled "Max NICs / Expected network bandwidth (Mbps)."

The throughput limit applies to the virtual machine. Throughput is not affected by these factors:

- **Number of network interfaces:** The bandwidth limit applies to the sum of all outbound traffic from the virtual machine.
- **Accelerated networking:** Though this feature can be helpful in achieving the published limit, it doesn't change the limit.
- **Traffic destination:** All destinations count toward the outbound limit.
- **Protocol:** All outbound traffic over all protocols counts towards the limit.

For more information, see [Virtual machine network bandwidth](#).

## Internet performance considerations

As discussed throughout this article, factors on the internet and outside the control of Azure can affect network performance. Here are some of those factors:

- **Latency:** The round-trip time between two destinations can be affected by issues on intermediate networks, by traffic that doesn't take the "shortest" distance path, and by suboptimal peering paths.
- **Packet loss:** Packet loss can be caused by network congestion, physical path issues, and underperforming network devices.
- **MTU size/Fragmentation:** Fragmentation along the path can lead to delays in data arrival or in packets arriving out of order, which can affect the delivery of packets.

Traceroute is a good tool for measuring network performance characteristics (like packet loss and latency) along every network path between a source device and a destination device.

## Network design considerations

Along with the considerations discussed earlier in this article, the topology of a virtual network can affect the network's performance. For example, a hub-and-spoke design that backhauls traffic globally to a single-hub virtual network will introduce network latency, which will affect overall network performance.

The number of network devices that network traffic passes through can also affect overall latency. For example, in a hub-and-spoke design, if traffic passes through a spoke network virtual appliance and a hub virtual appliance before transiting to the internet, the network virtual appliances can introduce latency.

## Azure regions, virtual networks, and latency

Azure regions are made up of multiple datacenters that exist within a general geographic area. These datacenters might not be physically next to each other. In some cases they're separated by as much as 10 kilometers. The virtual network is a logical overlay on top of the Azure physical datacenter network. A virtual network doesn't imply any specific network topology within the datacenter.

For example, two VMs that are in the same virtual network and subnet might be in different racks, rows, or even datacenters. They could be separated by feet of fiber optic cable or by kilometers of fiber optic cable. This

variation could introduce variable latency (a few milliseconds difference) between different VMs.

The geographic placement of VMs, and the potential resulting latency between two VMs, can be influenced by the configuration of availability sets and Availability Zones. But the distance between datacenters in a region is region-specific and primarily influenced by datacenter topology in the region.

### Source NAT port exhaustion

A deployment in Azure can communicate with endpoints outside of Azure on the public internet and/or in the public IP space. When an instance initiates an outbound connection, Azure dynamically maps the private IP address to a public IP address. After Azure creates this mapping, return traffic for the outbound originated flow can also reach the private IP address where the flow originated.

For every outbound connection, the Azure Load Balancer needs to maintain this mapping for some period of time. With the multitenant nature of Azure, maintaining this mapping for every outbound flow for every VM can be resource intensive. So there are limits that are set and based on the configuration of the Azure Virtual Network. Or, to say that more precisely, an Azure VM can only make a certain number of outbound connections at a given time. When these limits are reached, the VM won't be able to make more outbound connections.

But this behavior is configurable. For more information about SNAT and SNAT port exhaustion, see [this article](#).

## Measure network performance on Azure

A number of the performance maximums in this article are related to the network latency / round-trip time (RTT) between two VMs. This section provides some suggestions for how to test latency/RTT and how to test TCP performance and VM network performance. You can tune and performance test the TCP/IP and network values discussed earlier by using the techniques described in this section. You can plug latency, MTU, MSS, and window size values into the calculations provided earlier and compare theoretical maximums to actual values that you observe during testing.

### Measure round-trip time and packet loss

TCP performance relies heavily on RTT and packet Loss. The PING utility available in Windows and Linux provides the easiest way to measure RTT and packet loss. The output of PING will show the minimum/maximum/average latency between a source and destination. It will also show packet loss. PING uses the ICMP protocol by default. You can use PsPing to test TCP RTT. For more information, see [PsPing](#).

### Measure actual throughput of a TCP connection

NTttcp is a tool for testing the TCP performance of a Linux or Windows VM. You can change various TCP settings and then test the benefits by using NTttcp. For more information, see these resources:

- [Bandwidth/Throughput testing \(NTttcp\)](#)
- [NTttcp Utility](#)

### Measure actual bandwidth of a virtual machine

You can test the performance of different VM types, accelerated networking, and so on, by using a tool called iPerf. iPerf is also available on Linux and Windows. iPerf can use TCP or UDP to test overall network throughput. iPerf TCP throughput tests are influenced by the factors discussed in this article (like latency and RTT). So UDP might yield better results if you just want to test maximum throughput.

For more information, see these articles:

- [Troubleshooting Expressroute network performance](#)
- [How to validate VPN throughput to a virtual network](#)

### Detect inefficient TCP behaviors

In packet captures, Azure customers might see TCP packets with TCP flags (SACK, DUP ACK, RETRANSMIT, and

FAST RETRANSMIT) that could indicate network performance problems. These packets specifically indicate network inefficiencies that result from packet loss. But packet loss isn't necessarily caused by Azure performance problems. Performance problems could be the result of application problems, operating system problems, or other problems that might not be directly related to the Azure platform.

Also, keep in mind that some retransmission and duplicate ACKs are normal on a network. TCP protocols were built to be reliable. Evidence of these TCP packets in a packet capture doesn't necessarily indicate a systemic network problem, unless they're excessive.

Still, these packet types are indications that TCP throughput isn't achieving its maximum performance, for reasons discussed in other sections of this article.

## Next steps

Now that you've learned about TCP/IP performance tuning for Azure VMs, you might want to read about other considerations for [planning virtual networks](#) or [learn more about connecting and configuring virtual networks](#).

# Virtual machine network bandwidth

9/21/2022 • 3 minutes to read • [Edit Online](#)

Azure offers a variety of VM sizes and types, each with a different mix of performance capabilities. One capability is network throughput (or bandwidth), measured in megabits per second (Mbps). Because virtual machines are hosted on shared hardware, the network capacity must be shared fairly among the virtual machines sharing the same hardware. Larger virtual machines are allocated relatively more bandwidth than smaller virtual machines.

The network bandwidth allocated to each virtual machine is metered on egress (outbound) traffic from the virtual machine. All network traffic leaving the virtual machine is counted toward the allocated limit, regardless of destination. For example, if a virtual machine has a 1,000 Mbps limit, that limit applies whether the outbound traffic is destined for another virtual machine in the same virtual network, or outside of Azure.

Ingress is not metered or limited directly. However, there are other factors, such as CPU and storage limits, which can impact a virtual machine's ability to process incoming data.

Accelerated networking is a feature designed to improve network performance, including latency, throughput, and CPU utilization. While accelerated networking can improve a virtual machine's throughput, it can do so only up to the virtual machine's allocated bandwidth. To learn more about Accelerated networking, see Accelerated networking for [Windows](#) or [Linux](#) virtual machines.

Azure virtual machines must have one, but may have several, network interfaces attached to them. Bandwidth allocated to a virtual machine is the sum of all outbound traffic across all network interfaces attached to a virtual machine. In other words, the allocated bandwidth is per virtual machine, regardless of how many network interfaces are attached to the virtual machine. To learn how many network interfaces different Azure VM sizes support, see Azure [Windows](#) and [Linux](#) VM sizes.

## Expected network throughput

Expected outbound throughput and the number of network interfaces supported by each VM size is detailed in Azure [Windows](#) and [Linux](#) VM sizes. Select a type, such as General purpose, then select a size-series on the resulting page, such as the Dv2-series. Each series has a table with networking specifications in the last column titled, **Max NICs / Expected network performance (Mbps)**.

The throughput limit applies to the virtual machine. Throughput is unaffected by the following factors:

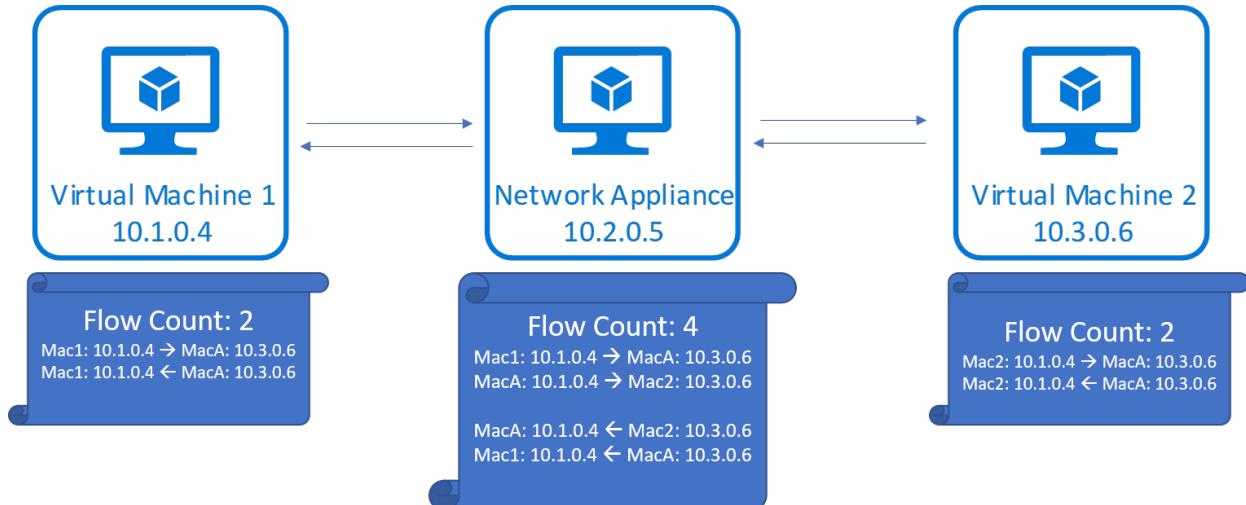
- **Number of network interfaces:** The bandwidth limit is cumulative of all outbound traffic from the virtual machine.
- **Accelerated networking:** Though the feature can be helpful in achieving the published limit, it does not change the limit.
- **Traffic destination:** All destinations count toward the outbound limit.
- **Protocol:** All outbound traffic over all protocols counts towards the limit.

## Network Flow Limits

In addition to bandwidth, the number of network connections present on a VM at any given time can affect its network performance. The Azure networking stack maintains state for each direction of a TCP/UDP connection in data structures called 'flows'. A typical TCP/UDP connection will have 2 flows created, one for the inbound and another for the outbound direction.

Data transfer between endpoints requires creation of several flows in addition to those that perform the data

transfer. Some examples are flows created for DNS resolution and flows created for load balancer health probes. Also note that network virtual appliances (NVAs) such as gateways, proxies, firewalls, will see flows being created for connections terminated at the appliance and originated by the appliance.



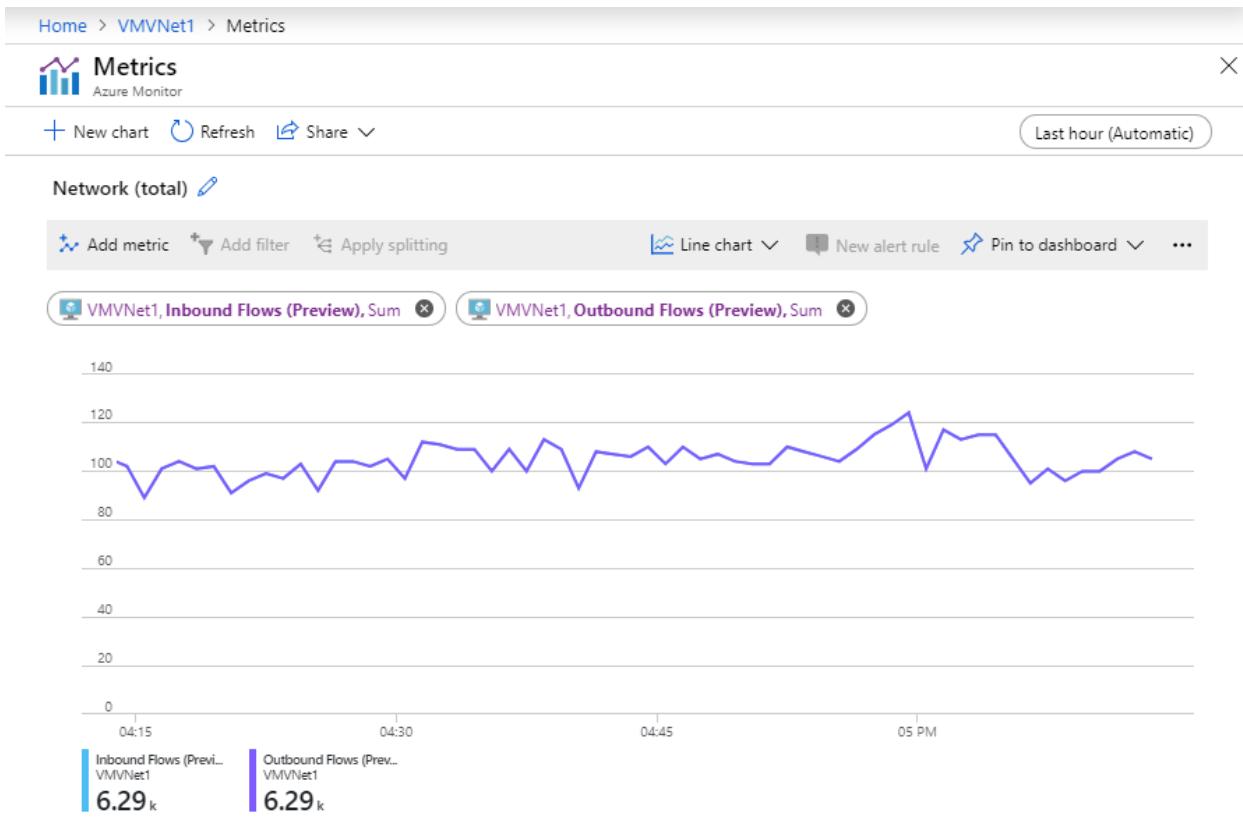
## Flow Limits and Active Connections Recommendations

Today, the Azure networking stack supports 1M total flows (500k inbound and 500k outbound) for a VM. Total active connections that can be handled by a VM in different scenarios are as follows.

- VMs that belong to VNET can handle 500k **active connections** for all VM sizes with 500k **active flows in each direction**.
- VMs with network virtual appliances (NVAs) such as gateway, proxy, firewall can handle 250k **active connections** with 500k **active flows in each direction** due to the forwarding and additional new flow creation on new connection setup to the next hop as shown in the above diagram.

Once this limit is hit, additional connections are dropped. Connection establishment and termination rates can also affect network performance as connection establishment and termination shares CPU with packet processing routines. We recommend that you benchmark workloads against expected traffic patterns and scale out workloads appropriately to match your performance needs.

Metrics are available in [Azure Monitor](#) to track the number of network flows and the flow creation rate on your VM or VMSS instances.



## Next steps

- Optimize network throughput for a virtual machine operating system
- Test network throughput for a virtual machine.

# Quickstart: Create and configure Azure DDoS Protection Standard

9/21/2022 • 5 minutes to read • [Edit Online](#)

Get started with Azure DDoS Protection Standard by using the Azure portal.

A DDoS protection plan defines a set of virtual networks that have DDoS Protection Standard enabled, across subscriptions. You can configure one DDoS protection plan for your organization and link virtual networks from multiple subscriptions under a single AAD tenant to the same plan.

In this quickstart, you'll create a DDoS protection plan and link it to a virtual network.

## Prerequisites

- If you don't have an Azure subscription, create a [free account](#) before you begin.
- Sign in to the Azure portal at <https://portal.azure.com>. Ensure that your account is assigned to the [network contributor](#) role or to a [custom role](#) that is assigned the appropriate actions listed in the how-to guide on [Permissions](#).

## Create a DDoS protection plan

1. Select **Create a resource** in the upper left corner of the Azure portal.
2. Search the term *DDoS*. When **DDoS protection plan** appears in the search results, select it.
3. Select **Create**.
4. Enter or select the following values.

SETTING	VALUE
Subscription	Select your subscription.
Resource group	Select <b>Create new</b> and enter <b>MyResourceGroup</b> .
Name	Enter <b>MyDdosProtectionPlan</b> .
Region	Enter <b>East US</b> .

5. Select **Review + create** then **Create**

### NOTE

Although DDoS Protection Plan resources need to be associated with a region, users can enable DDoS protection on Virtual Networks in different regions and across multiple subscriptions under a single Azure Active Directory Tenant.

## Enable DDoS protection for a virtual network

### Enable DDoS protection for a new virtual network

1. Select **Create a resource** in the upper left corner of the Azure portal.

2. Select **Networking**, and then select **Virtual network**.

3. Enter or select the following values.

SETTING	VALUE
Subscription	Select your subscription.
Resource group	Select <b>Use existing</b> , and then select <b>MyResourceGroup</b>
Name	Enter <b>MyVnet</b> .
Region	Enter <b>East US</b> .

4. Select **Next: IP Addresses** and enter the following values.

SETTING	VALUE
IPv4 address space	Enter <b>10.1.0.0/16</b> .
Subnet name	Under <b>Subnet name</b> , select the <b>Add subnet</b> link and enter <b>mySubnet</b> .
Subnet address range	Enter <b>10.1.0.0/24</b> .

5. Select **Add**.

6. Select **Next: Security**.

7. Select **Enable** on the **DDoS Protection Standard** radio.

8. Select **MyDdosProtectionPlan** from the **DDoS protection plan** pane. The plan you select can be in the same, or different subscription than the virtual network, but both subscriptions must be associated to the same Azure Active Directory tenant.

9. Select **Review + create** then **Create**.

#### NOTE

You cannot move a virtual network to another resource group or subscription when DDoS Standard is enabled for the virtual network. If you need to move a virtual network with DDoS Standard enabled, disable DDoS Standard first, move the virtual network, and then enable DDoS standard. After the move, the auto-tuned policy thresholds for all the protected public IP addresses in the virtual network are reset.

#### Enable DDoS protection for an existing virtual network

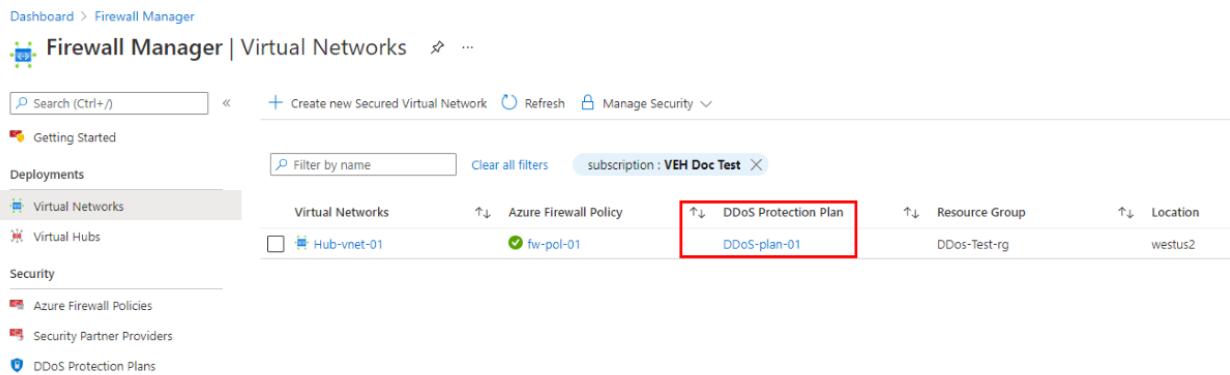
1. Create a DDoS protection plan by completing the steps in [Create a DDoS protection plan](#), if you don't have an existing DDoS protection plan.
2. Enter the name of the virtual network that you want to enable DDoS Protection Standard for in the **Search resources, services, and docs** box at the top of the Azure portal. When the name of the virtual network appears in the search results, select it.
3. Select **DDoS protection**, under **Settings**.
4. Select **Enable**. Under **DDoS protection plan**, select an existing DDoS protection plan, or the plan you created in step 1, and then click **Save**. The plan you select can be in the same, or different subscription than the virtual network, but both subscriptions must be associated to the same Azure Active Directory tenant.

You can also enable the DDoS protection plan for an existing virtual network from the DDoS Protection plan, not from the virtual network.

1. Search for "DDoS protection plans" in the **Search resources, services, and docs** box at the top of the Azure portal. When **DDoS protection plans** appears in the search results, select it.
2. Select the desired DDoS protection plan you want to enable for your virtual network.
3. Select **Protected resources** under **Settings**.
4. Click **+Add** and select the right subscription, resource group and the virtual network name. Click **Add** again.

## Configure an Azure DDoS Protection Plan using Azure Firewall Manager (preview)

Azure Firewall Manager is a platform to manage and protect your network resources at scale. You can associate your virtual networks with a DDoS protection plan within Azure Firewall Manager. This functionality is currently available in Public Preview. See [Configure an Azure DDoS Protection Plan using Azure Firewall Manager](#).



The screenshot shows the Azure Firewall Manager interface. On the left, there's a sidebar with links like 'Getting Started', 'Deployments', 'Virtual Networks' (which is selected and highlighted in grey), 'Virtual Hubs', 'Security', 'Azure Firewall Policies', 'Security Partner Providers', and 'DDoS Protection Plans'. The main area has a header with 'Search (Ctrl+J)', '+ Create new Secured Virtual Network', 'Refresh', and 'Manage Security'. Below the header, there's a filter bar with 'Filter by name' and 'Clear all filters', and a dropdown for 'subscription : VEH Doc Test'. The main table lists virtual networks with columns: 'Virtual Networks' (containing 'Hub-vnet-01'), 'Azure Firewall Policy' (containing 'fw-pol-01'), 'DDoS Protection Plan' (containing 'DDoS-plan-01' which is highlighted with a red box), 'Resource Group' (containing 'Ddos-Test-rg'), and 'Location' (containing 'westus2').

## Enable DDoS protection for all virtual networks

This [built-in policy](#) will detect any virtual networks in a defined scope that don't have DDoS Protection Standard enabled. This policy will then optionally create a remediation task that will create the association to protect the Virtual Network. See [Azure Policy built-in definitions for Azure DDoS Protection Standard](#) for full list of built-in policies.

## Validate and test

First, check the details of your DDoS protection plan:

1. Select **All services** on the top, left of the portal.
2. Enter **DDoS** in the **Filter** box. When **DDoS protection plans** appear in the results, select it.
3. Select your DDoS protection plan from the list.

The *MyVnet* virtual network should be listed.

## View protected resources

Under **Protected resources**, you can view your protected virtual networks and public IP addresses, or add more virtual networks to your DDoS protection plan:

The screenshot shows the Azure portal interface for managing a DDoS protection plan. The left sidebar has a tree view with 'Protected resources' selected under 'ddosplanav | Protected resources'. The main area displays a table of resources with columns: Public IP address, Virtual network, Application gateway, and Subscription. The table shows four entries: myAppGatewayDDoSCTest, NewDDoSvNET, myAppGatewayDDoS, Azure DDoS Protection Demo Sub; newddosappgwip, ddostest, ddosbillingappgw, Azure DDoS Protection Demo Sub; ddostest, appgw-vnet-wcu, appgw-wcu, Azure DDoS Protection Demo Sub; and appgw-wcu, appgw-wcu, appgw-wcu, Azure DDoS Protection Demo Sub.

## Clean up resources

You can keep your resources for the next tutorial. If no longer needed, delete the *MyResourceGroup* resource group. When you delete the resource group, you also delete the DDoS protection plan and all its related resources. If you don't intend to use this DDoS protection plan, you should remove resources to avoid unnecessary charges.

### WARNING

This action is irreversible.

1. In the Azure portal, search for and select **Resource groups**, or select **Resource groups** from the Azure portal menu.
2. Filter or scroll down to find the *MyResourceGroup* resource group.
3. Select the resource group, then select **Delete resource group**.
4. Type the resource group name to verify, and then select **Delete**.

To disable DDoS protection for a virtual network:

1. Enter the name of the virtual network you want to disable DDoS protection standard for in the **Search resources, services, and docs** box at the top of the portal. When the name of the virtual network appears in the search results, select it.
2. Under **DDoS Protection Standard**, select **Disable**.

### NOTE

If you want to delete a DDoS protection plan, you must first dissociate all virtual networks from it.

## Next steps

To learn how to view and configure telemetry for your DDoS protection plan, continue to the tutorials.

[View and configure DDoS protection telemetry](#)

# Quickstart: Create and configure Azure DDoS Protection Standard using Azure PowerShell

9/21/2022 • 3 minutes to read • [Edit Online](#)

Get started with Azure DDoS Protection Standard by using Azure PowerShell.

A DDoS protection plan defines a set of virtual networks that have DDoS protection standard enabled, across subscriptions. You can configure one DDoS protection plan for your organization and link virtual networks from multiple subscriptions to the same plan.

In this quickstart, you'll create a DDoS protection plan and link it to a virtual network.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- Azure PowerShell installed locally or Azure Cloud Shell

### NOTE

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az](#).

## Azure Cloud Shell

Azure hosts Azure Cloud Shell, an interactive shell environment that you can use through your browser. You can use either Bash or PowerShell with Cloud Shell to work with Azure services. You can use the Cloud Shell preinstalled commands to run the code in this article, without having to install anything on your local environment.

To start Azure Cloud Shell:

OPTION	EXAMPLE/LINK
Select Try It in the upper-right corner of a code or command block. Selecting Try It doesn't automatically copy the code or command to Cloud Shell.	
Go to <a href="https://shell.azure.com">https://shell.azure.com</a> , or select the Launch Cloud Shell button to open Cloud Shell in your browser.	
Select the Cloud Shell button on the menu bar at the upper right in the <a href="#">Azure portal</a> .	

To use Azure Cloud Shell:

1. Start Cloud Shell.
2. Select the **Copy** button on a code block (or command block) to copy the code or command.
3. Paste the code or command into the Cloud Shell session by selecting **Ctrl+Shift+V** on Windows and Linux, or by selecting **Cmd+Shift+V** on macOS.

4. Select Enter to run the code or command.

## Create a DDoS Protection plan

In Azure, you allocate related resources to a resource group. You can either use an existing resource group or create a new one.

To create a resource group, use [New-AzResourceGroup](#). In this example, we'll name our resource group *MyResourceGroup* and use the *East US* location:

```
New-AzResourceGroup -Name MyResourceGroup -Location "East US"
```

Now create a DDoS protection plan named *MyDdosProtectionPlan*:

```
New-AzDdosProtectionPlan -ResourceGroupName MyResourceGroup -Name MyDdosProtectionPlan -Location "East US"
```

## Enable DDoS for a virtual network

### Enable DDoS for a new virtual network

You can enable DDoS protection when creating a virtual network. In this example, we'll name our virtual network *MyVnet*.

```
#Gets the DDoS protection plan ID  
$ddosProtectionPlanID = Get-AzDdosProtectionPlan -ResourceGroupName MyResourceGroup -Name  
MyDdosProtectionPlan  
  
#Creates the virtual network  
New-AzVirtualNetwork -Name MyVnet -ResourceGroupName MyResourceGroup -Location "East US" -AddressPrefix  
10.0.0.0/16 -DdosProtectionPlan $ddosProtectionPlanID -EnableDdosProtection
```

### Enable DDoS for an existing virtual network

You can associate an existing virtual network when creating a DDoS protection plan:

```
#Gets the DDoS protection plan ID  
$ddosProtectionPlanID = Get-AzDdosProtectionPlan -ResourceGroupName MyResourceGroup -Name  
MyDdosProtectionPlan  
  
# Gets the most updated version of the virtual network  
$vnet = Get-AzVirtualNetwork -Name MyVnet -ResourceGroupName MyResourceGroup  
$vnet.DdosProtectionPlan = New-Object Microsoft.Azure.Commands.Network.Models.PSResourceId  
  
# Update the properties and enable DDoS protection  
$vnet.DdosProtectionPlan.Id = $ddosProtectionPlanID.Id  
$vnet.EnableDdosProtection = $true  
$vnet | Set-AzVirtualNetwork
```

## Validate and test

Check the details of your DDoS protection plan and verify that the command returns the correct details of your DDoS protection plan.

```
Get-AzDdosProtectionPlan -ResourceGroupName MyResourceGroup -Name MyDdosProtectionPlan
```

Check the details of your vNet and verify the DDoS protection plan is enabled.

```
Get-AzVirtualNetwork -Name MyVnet -ResourceGroupName MyResourceGroup
```

## Clean up resources

You can keep your resources for the next tutorial. If no longer needed, delete the *MyResourceGroup* resource group. When you delete the resource group, you also delete the DDoS protection plan and all its related resources.

```
Remove-AzResourceGroup -Name MyResourceGroup
```

To disable DDoS protection for a virtual network:

```
# Gets the most updated version of the virtual network
$vnet = Get-AzVirtualNetwork -Name MyVnet -ResourceGroupName MyResourceGroup
$vnet.DdosProtectionPlan = $null
$vnet.EnableDdosProtection = $false
$vnet | Set-AzVirtualNetwork
```

If you want to delete a DDoS protection plan, you must first dissociate all virtual networks from it.

## Next steps

To learn how to view and configure telemetry for your DDoS protection plan, continue to the tutorials.

[View and configure DDoS protection telemetry](#)

# Quickstart: Create and configure Azure DDoS Protection Standard using Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

Get started with Azure DDoS Protection Standard by using Azure CLI.

A DDoS protection plan defines a set of virtual networks that have DDoS protection standard enabled, across subscriptions. You can configure one DDoS protection plan for your organization and link virtual networks from multiple subscriptions to the same plan.

In this quickstart, you'll create a DDoS protection plan and link it to a virtual network.

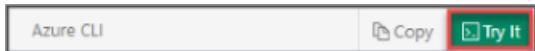
## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- Azure CLI installed locally or Azure Cloud Shell

## Azure Cloud Shell

Azure hosts Azure Cloud Shell, an interactive shell environment that you can use through your browser. You can use either Bash or PowerShell with Cloud Shell to work with Azure services. You can use the Cloud Shell preinstalled commands to run the code in this article, without having to install anything on your local environment.

To start Azure Cloud Shell:

OPTION	EXAMPLE/LINK
Select Try It in the upper-right corner of a code or command block. Selecting Try It doesn't automatically copy the code or command to Cloud Shell.	
Go to <a href="https://shell.azure.com">https://shell.azure.com</a> , or select the Launch Cloud Shell button to open Cloud Shell in your browser.	
Select the Cloud Shell button on the menu bar at the upper right in the <a href="#">Azure portal</a> .	

To use Azure Cloud Shell:

1. Start Cloud Shell.
2. Select the **Copy** button on a code block (or command block) to copy the code or command.
3. Paste the code or command into the Cloud Shell session by selecting **Ctrl+Shift+V** on Windows and Linux, or by selecting **Cmd+Shift+V** on macOS.
4. Select **Enter** to run the code or command.

If you choose to install and use the CLI locally, this quickstart requires Azure CLI version 2.0.56 or later. To find the version, run `az --version`. If you need to install or upgrade, see [Install the Azure CLI](#).

# Create a DDoS Protection plan

In Azure, you allocate related resources to a resource group. You can either use an existing resource group or create a new one.

To create a resource group, use [az group create](#). In this example, we'll name our resource group *MyResourceGroup* and use the *East US* location:

```
az group create \
--name MyResourceGroup \
--location eastus
```

Now create a DDoS protection plan named *MyDdosProtectionPlan*:

```
az network ddos-protection create \
--resource-group MyResourceGroup \
--name MyDdosProtectionPlan
```

## Enable DDoS protection for a virtual network

### Enable DDoS protection for a new virtual network

You can enable DDoS protection when creating a virtual network. In this example, we'll name our virtual network *MyVnet*.

```
az network vnet create \
--resource-group MyResourceGroup \
--name MyVnet \
--location eastus \
--ddos-protection-plan MyDdosProtectionPlan \
--ddos-protection true
```

#### NOTE

You cannot move a virtual network to another resource group or subscription when DDoS Standard is enabled for the virtual network. If you need to move a virtual network with DDoS Standard enabled, disable DDoS Standard first, move the virtual network, and then enable DDoS standard. After the move, the auto-tuned policy thresholds for all the protected public IP addresses in the virtual network are reset.

### Enable DDoS protection for an existing virtual network

When [creating a DDoS protection plan](#), you can associate one or more virtual networks to the plan. To add more than one virtual network, simply list the names or IDs, space-separated. In this example, we'll add *MyVnet*.

```
az group create \
--name MyResourceGroup \
--location eastus

az network ddos-protection create \
--resource-group MyResourceGroup \
--name MyDdosProtectionPlan \
--vnets MyVnet
```

Alternatively, you can enable DDoS protection for a given virtual network:

```
az network vnet update \
--resource-group MyResourceGroup \
--name MyVnet \
--ddos-protection-plan MyDdosProtectionPlan \
--ddos-protection true
```

## Validate and test

First, check the details of your DDoS protection plan:

```
az network ddos-protection show \
--resource-group MyResourceGroup \
--name MyDdosProtectionPlan
```

Verify that the command returns the correct details of your DDoS protection plan.

## Clean up resources

You can keep your resources for the next tutorial. If no longer needed, delete the *MyResourceGroup* resource group. When you delete the resource group, you also delete the DDoS protection plan and all its related resources.

To delete the resource group use [az group delete](#):

```
az group delete \
--name MyResourceGroup
```

Update a given virtual network to disable DDoS protection:

```
az network vnet update \
--resource-group MyResourceGroup \
--name MyVnet \
--ddos-protection-plan MyDdosProtectionPlan \
--ddos-protection false
```

If you want to delete a DDoS protection plan, you must first dissociate all virtual networks from it.

## Next steps

To learn how to view and configure telemetry for your DDoS protection plan, continue to the tutorials.

[View and configure DDoS protection telemetry](#)

# Quickstart: Create an Azure DDoS Protection Standard using ARM template

9/21/2022 • 3 minutes to read • [Edit Online](#)

This quickstart describes how to use an Azure Resource Manager template (ARM template) to create a distributed denial of service (DDoS) protection plan and virtual network (VNet), then enables the protection plan for the VNet. An Azure DDoS Protection Standard plan defines a set of virtual networks that have DDoS protection enabled across subscriptions. You can configure one DDoS protection plan for your organization and link virtual networks from multiple subscriptions to the same plan.

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

If your environment meets the prerequisites and you're familiar with using ARM templates, select the **Deploy to Azure** button. The template will open in the Azure portal.



## Prerequisites

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Review the template

The template used in this quickstart is from [Azure Quickstart Templates](#).

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "metadata": {  
    "_generator": {  
      "name": "bicep",  
      "version": "0.5.6.12127",  
      "templateHash": "14909118711877377105"  
    }  
  },  
  "parameters": {  
    "ddosProtectionPlanName": {  
      "type": "string",  
      "metadata": {  
        "description": "Specify a DDoS protection plan name."  
      }  
    },  
    "virtualNetworkName": {  
      "type": "string",  
      "metadata": {  
        "description": "Specify a DDoS virtual network name."  
      }  
    },  
    "location": {  
      "type": "string",  
      "defaultValue": "[resourceGroup().location]",  
      "metadata": {  
        "description": "Specify a location for the resources."  
      }  
    }  
  }  
}
```

```

        },
        "vnetAddressPrefix": {
            "type": "string",
            "defaultValue": "172.17.0.0/16",
            "metadata": {
                "description": "Specify the virtual network address prefix"
            }
        },
        "subnetPrefix": {
            "type": "string",
            "defaultValue": "172.17.0.0/24",
            "metadata": {
                "description": "Specify the virtual network subnet prefix"
            }
        },
        "ddosProtectionPlanEnabled": {
            "type": "bool",
            "defaultValue": true,
            "metadata": {
                "description": "Enable DDoS protection plan."
            }
        }
    },
    "resources": [
        {
            "type": "Microsoft.Network/ddosProtectionPlans",
            "apiVersion": "2021-05-01",
            "name": "[parameters('ddosProtectionPlanName')]",
            "location": "[parameters('location')]"
        },
        {
            "type": "Microsoft.Network/virtualNetworks",
            "apiVersion": "2021-05-01",
            "name": "[parameters('virtualNetworkName')]",
            "location": "[parameters('location')]",
            "properties": {
                "addressSpace": {
                    "addressPrefixes": [
                        "[parameters('vnetAddressPrefix')]"
                    ]
                },
                "subnets": [
                    {
                        "name": "default",
                        "properties": {
                            "addressPrefix": "[parameters('subnetPrefix')]"
                        }
                    }
                ],
                "enableDdosProtection": "[parameters('ddosProtectionPlanEnabled')]",
                "ddosProtectionPlan": {
                    "id": "[resourceId('Microsoft.Network/ddosProtectionPlans',
parameters('ddosProtectionPlanName'))]"
                }
            },
            "dependsOn": [
                "[resourceId('Microsoft.Network/ddosProtectionPlans', parameters('ddosProtectionPlanName'))]"
            ]
        }
    ]
}

```

The template defines two resources:

- [Microsoft.Network/ddosProtectionPlans](#)
- [Microsoft.Network/virtualNetworks](#)

# Deploy the template

In this example, the template creates a new resource group, a DDoS protection plan, and a VNet.

1. To sign in to Azure and open the template, select the **Deploy to Azure** button.



2. Enter the values to create a new resource group, DDoS protection plan, and VNet name.

The screenshot shows the 'Create and enable a DDOS protection plan' template in the Azure portal. It's a 'Template' type with 2 resources. The 'Subscription' dropdown is set to 'Azure subscription name'. The 'Resource group' dropdown is set to '(New) MyResourceGroup', which is highlighted with a red box. The 'Region' dropdown is set to 'East US'. The 'Ddos Protection Plan Name' dropdown is set to 'MyDdosProtectionPlan'. The 'Virtual Network Name' dropdown is set to 'MyVNet'. Other fields like 'Location', 'Vnet Address Prefix', 'Subnet Prefix', and 'Ddos Protection Plan Enabled' are also visible. At the bottom, there are 'Review + create' and 'Next : Review + create >' buttons.

- **Subscription:** Name of the Azure subscription where the resources will be deployed.
- **Resource group:** Select an existing resource group or create a new resource group.
- **Region:** The region where the resource group is deployed, such as East US.
- **Ddos Protection Plan Name:** The name of for the new DDoS protection plan.
- **Virtual Network Name:** Creates a name for the new VNet.
- **Location:** Function that uses the same region as the resource group for resource deployment.
- **Vnet Address Prefix:** Use the default value or enter your VNet address.
- **Subnet Prefix:** Use the default value or enter your VNet subnet.
- **Ddos Protection Plan Enabled:** Default is `true` to enable the DDoS protection plan.

3. Select **Review + create**.

4. Verify that template validation passed and select **Create** to begin the deployment.

## Review deployed resources

To copy the Azure CLI or Azure PowerShell command, select the **Copy** button. The **Try it** button opens Azure Cloud Shell to run the command.

- [CLI](#)
- [PowerShell](#)

```
az network ddos-protection show \
--resource-group MyResourceGroup \
--name MyDdosProtectionPlan
```

The output shows the new resources.

- [CLI](#)
- [PowerShell](#)

```
{
  "etag": "W/\"abcdefg-1111-2222-bbbb-987654321098\"",
  "id": "/subscriptions/b1111111-2222-3333-aaaa-
012345678912/resourceGroups/MyResourceGroup/providers/Microsoft.Network/ddosProtectionPlans/MyDdosProtection
Plan",
  "location": "eastus",
  "name": "MyDdosProtectionPlan",
  "provisioningState": "Succeeded",
  "resourceGroup": "MyResourceGroup",
  "resourceGuid": null,
  "tags": null,
  "type": "Microsoft.Network/ddosProtectionPlans",
  "virtualNetworks": [
    {
      "id": "/subscriptions/b1111111-2222-3333-aaaa-
012345678912/resourceGroups/MyResourceGroup/providers/Microsoft.Network/virtualNetworks/MyVNet",
      "resourceGroup": "MyResourceGroup"
    }
  ]
}
```

## Clean up resources

When you're finished you can delete the resources. The command deletes the resource group and all the resources it contains.

- [CLI](#)
- [PowerShell](#)

```
az group delete --name MyResourceGroup
```

## Next steps

To learn how to view and configure telemetry for your DDoS protection plan, continue to the tutorials.

[View and configure DDoS protection telemetry](#)

# Open ports and endpoints to a VM with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

You open a port, or create an endpoint, to a virtual machine (VM) in Azure by creating a network filter on a subnet or VM network interface. You place these filters, which control both inbound and outbound traffic, on a Network Security Group attached to the resource that receives the traffic. Let's use a common example of web traffic on port 80. This article shows you how to open a port to a VM with the Azure CLI.

To create a Network Security Group and rules you need the latest [Azure CLI](#) installed and logged in to an Azure account using [az login](#).

In the following examples, replace example parameter names with your own values. Example parameter names include *myResourceGroup*, *myNetworkSecurityGroup*, and *myVnet*.

## Quickly open a port for a VM

If you need to quickly open a port for a VM in a dev/test scenario, you can use the [az vm open-port](#) command. This command creates a Network Security Group, adds a rule, and applies it to a VM or subnet. The following example opens port *80* on the VM named *myVM* in the resource group named *myResourceGroup*.

```
az vm open-port --resource-group myResourceGroup --name myVM --port 80
```

For more control over the rules, such as defining a source IP address range, continue with the additional steps in this article.

## Create a Network Security Group and rules

Create the network security group with [az network nsg create](#). The following example creates a network security group named *myNetworkSecurityGroup* in the *eastus* location:

```
az network nsg create \
--resource-group myResourceGroup \
--location eastus \
--name myNetworkSecurityGroup
```

Add a rule with [az network nsg rule create](#) to allow HTTP traffic to your webserver (or adjust for your own scenario, such as SSH access or database connectivity). The following example creates a rule named *myNetworkSecurityGroupRule* to allow TCP traffic on port 80:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name myNetworkSecurityGroupRule \
--protocol tcp \
--priority 1000 \
--destination-port-range 80
```

## Apply Network Security Group to VM

Associate the Network Security Group with your VM's network interface (NIC) with [az network nic update](#). The following example associates an existing NIC named *myNic* with the Network Security Group named *myNetworkSecurityGroup*.

```
az network nic update \
--resource-group myResourceGroup \
--name myNic \
--network-security-group myNetworkSecurityGroup
```

Alternatively, you can associate your Network Security Group with a virtual network subnet with [az network vnet subnet update](#) rather than just to the network interface on a single VM. The following example associates an existing subnet named *mySubnet* in the *myVnet* virtual network with the Network Security Group named *myNetworkSecurityGroup*.

```
az network vnet subnet update \
--resource-group myResourceGroup \
--vnet-name myVnet \
--name mySubnet \
--network-security-group myNetworkSecurityGroup
```

## More information on Network Security Groups

The quick commands here allow you to get up and running with traffic flowing to your VM. Network Security Groups provide many great features and granularity for controlling access to your resources. You can read more about [creating a Network Security Group and ACL rules here](#).

For highly available web applications, you should place your VMs behind an Azure Load Balancer. The load balancer distributes traffic to VMs, with a Network Security Group that provides traffic filtering. For more information, see [How to load balance Linux virtual machines in Azure to create a highly available application](#).

## Next steps

In this example, you created a simple rule to allow HTTP traffic. You can find information on creating more detailed environments in the following articles:

- [Azure Resource Manager overview](#)
- [What is a Network Security Group \(NSG\)?](#)

# How to open ports and endpoints to a VM using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

You open a port, or create an endpoint, to a virtual machine (VM) in Azure by creating a network filter on a subnet or a VM network interface. You place these filters, which control both inbound and outbound traffic, on a network security group attached to the resource that receives the traffic.

The example in this article demonstrates how to create a network filter that uses the standard TCP port 80 (it's assumed you've already started the appropriate services and opened any OS firewall rules on the VM).

After you've created a VM that's configured to serve web requests on the standard TCP port 80, you can:

1. Create a network security group.
2. Create an inbound security rule allowing traffic and assign values to the following settings:
  - **Destination port ranges:** 80
  - **Source port ranges:** \* (allows any source port)
  - **Priority value:** Enter a value that is less than 65,500 and higher in priority than the default catch-all deny inbound rule.
3. Associate the network security group with the VM network interface or subnet.

Although this example uses a simple rule to allow HTTP traffic, you can also use network security groups and rules to create more complex network configurations.

## Quick commands

To create a Network Security Group and ACL rules you need [the latest version of Azure PowerShell installed](#). You can also [perform these steps using the Azure portal](#).

Log in to your Azure account:

```
Connect-AzAccount
```

In the following examples, replace parameter names with your own values. Example parameter names included *myResourceGroup*, *myNetworkSecurityGroup*, and *myVnet*.

Create a rule with [New-AzNetworkSecurityRuleConfig](#). The following example creates a rule named *myNetworkSecurityGroupRule* to allow *tcp* traffic on port *80*:

```
$httprule = New-AzNetworkSecurityRuleConfig `  
    -Name "myNetworkSecurityGroupRule" `  
    -Description "Allow HTTP" `  
    -Access "Allow" `  
    -Protocol "Tcp" `  
    -Direction "Inbound" `  
    -Priority 100 `  
    -SourceAddressPrefix "Internet" `  
    -SourcePortRange * `  
    -DestinationAddressPrefix * `  
    -DestinationPortRange "80"
```

Next, create your Network Security group with [New-AzNetworkSecurityGroup](#) and assign the HTTP rule you just created as follows. The following example creates a Network Security Group named *myNetworkSecurityGroup*:

```
$nsg = New-AzNetworkSecurityGroup `  
    -ResourceGroupName "myResourceGroup" `  
    -Location "EastUS" `  
    -Name "myNetworkSecurityGroup" `  
    -SecurityRules $httprule
```

Now let's assign your Network Security Group to a subnet. The following example assigns an existing virtual network named *myVnet* to the variable `$vnet` with [Get-AzVirtualNetwork](#):

```
$vnet = Get-AzVirtualNetwork `  
    -ResourceGroupName "myResourceGroup" `  
    -Name "myVnet"
```

Associate your Network Security Group with your subnet with [Set-AzVirtualNetworkSubnetConfig](#). The following example associates the subnet named *mySubnet* with your Network Security Group:

```
$subnetPrefix = $vnet.Subnets | ?{$_ . Name -eq 'mySubnet'}`  
  
Set-AzVirtualNetworkSubnetConfig `  
    -VirtualNetwork $vnet `  
    -Name "mySubnet" `  
    -AddressPrefix $subnetPrefix.AddressPrefix `  
    -NetworkSecurityGroup $nsg
```

Finally, update your virtual network with [Set-AzVirtualNetwork](#) in order for your changes to take effect:

```
Set-AzVirtualNetwork -VirtualNetwork $vnet
```

## More information on Network Security Groups

The quick commands here allow you to get up and running with traffic flowing to your VM. Network Security Groups provide many great features and granularity for controlling access to your resources. You can read more about [creating a Network Security Group and ACL rules here](#).

For highly available web applications, you should place your VMs behind an Azure Load Balancer. The load balancer distributes traffic to VMs, with a Network Security Group that provides traffic filtering. For more information, see [How to load balance Linux virtual machines in Azure to create a highly available application](#).

## Next steps

In this example, you created a simple rule to allow HTTP traffic. You can find information on creating more detailed environments in the following articles:

- [Azure Resource Manager overview](#)
- [What is a network security group?](#)
- [Azure Load Balancer Overview](#)

# How to open ports to a virtual machine with the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

You open a port, or create an endpoint, to a virtual machine (VM) in Azure by creating a network filter on a subnet or a VM network interface. You place these filters, which control both inbound and outbound traffic, on a network security group attached to the resource that receives the traffic.

The example in this article demonstrates how to create a network filter that uses the standard TCP port 80 (it's assumed you've already started the appropriate services and opened any OS firewall rules on the VM).

After you've created a VM that's configured to serve web requests on the standard TCP port 80, you can:

1. Create a network security group.
2. Create an inbound security rule allowing traffic and assign values to the following settings:
  - **Destination port ranges:** 80
  - **Source port ranges:** \* (allows any source port)
  - **Priority value:** Enter a value that is less than 65,500 and higher in priority than the default catch-all deny inbound rule.
3. Associate the network security group with the VM network interface or subnet.

Although this example uses a simple rule to allow HTTP traffic, you can also use network security groups and rules to create more complex network configurations.

## Sign in to Azure

Sign in to the Azure portal at <https://portal.azure.com>.

## Create a network security group

1. Search for and select the resource group for the VM, choose **Add**, then search for and select **Network security group**.
2. Select **Create**.

The **Create network security group** window opens.

## Create network security group

Basics Tags Review + create

Project details

Subscription \* myAzureSubscription

Resource group \* myresourcegroup [Create new](#)

Instance details

Name \* myNSG

Region \* (US) East US

---

[Review + create](#) [< Previous](#) [Next : Tags >](#) [Download a template for automation](#)

3. Enter a name for your network security group.
4. Select or create a resource group, then select a location.
5. Select **Create** to create the network security group.

## Create an inbound security rule

1. Select your new network security group.
2. Select **Inbound security rules** from the left menu, then select **Add**.



## Add inbound security rule

X

Source ⓘ

Any

Source port ranges \* ⓘ

\*

Destination ⓘ

Any

Service ⓘ

HTTP

Destination port ranges ⓘ

80

Protocol

Any

TCP

UDP

ICMP

Action

Allow

Deny

Priority \* ⓘ

100

Name \*

Port\_8080

Description

**Add** **Cancel**

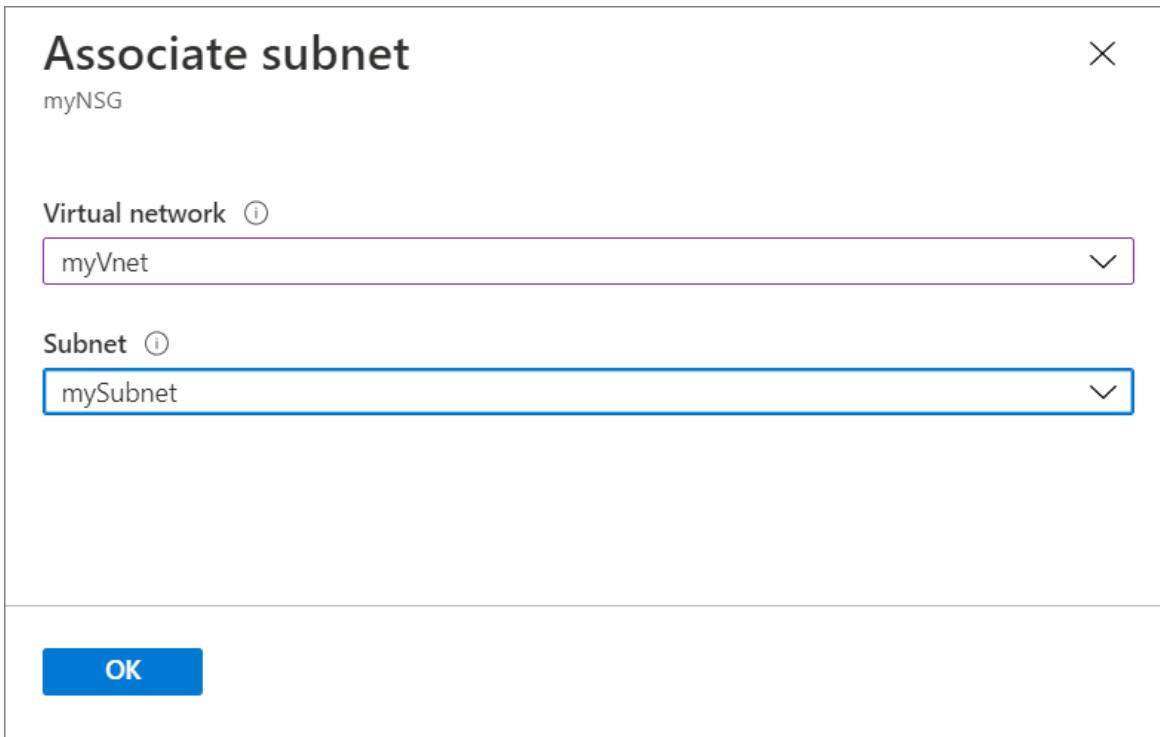
3. You can limit the **Source** and **Source port ranges** as needed or leave the default of *Any*.
4. You can limit the **Destination** as needed or leave the default of *Any*.
5. Choose a common **Service** from the drop-down menu, such as **HTTP**. You can also select **Custom** if you want to provide a specific port to use.
6. Optionally, change the **Priority** or **Name**. The priority affects the order in which rules are applied: the lower the numerical value, the earlier the rule is applied.
7. Select **Add** to create the rule.

## Associate your network security group with a subnet

Your final step is to associate your network security group with a subnet or a specific network interface. For this example, we'll associate the network security group with a subnet.

1. Select **Subnets** from the left menu, then select **Associate**.

2. Select your virtual network, and then select the appropriate subnet.



3. When you are done, select OK.

## Additional information

You can also [perform the steps in this article by using Azure PowerShell](#).

The commands described in this article allow you to quickly get traffic flowing to your VM. Network security groups provide many great features and granularity for controlling access to your resources. For more information, see [Filter network traffic with a network security group](#).

For highly available web applications, consider placing your VMs behind an Azure load balancer. The load balancer distributes traffic to VMs, with a network security group that provides traffic filtering. For more information, see [Load balance Windows virtual machines in Azure to create a highly available application](#).

## Next steps

In this article, you created a network security group, created an inbound rule that allows HTTP traffic on port 80, and then associated that rule with a subnet.

You can find information on creating more detailed environments in the following articles:

- [Azure Resource Manager overview](#)
- [Security groups](#)

# Tutorial: Migrate a virtual machine public IP address to Azure Virtual Network NAT

9/21/2022 • 4 minutes to read • [Edit Online](#)

In this article, you'll learn how to migrate your virtual machine's public IP address to a NAT gateway. You'll learn how to remove the IP address from the virtual machine. You'll reuse the IP address from the virtual machine for the NAT gateway.

Azure Virtual Network NAT is the recommended method for outbound connectivity. A NAT gateway is a fully managed and highly resilient Network Address Translation (NAT) service. A NAT gateway doesn't have the same limitations of SNAT port exhaustion as default outbound access. A NAT gateway replaces the need for a virtual machine to have a public IP address to have outbound connectivity.

For more information about Azure Virtual Network NAT, see [What is Azure Virtual Network NAT](#)

In this tutorial, you learn how to:

- Remove the public IP address from the virtual machine.
- Associate the public IP address from the virtual machine with a NAT gateway.

## Prerequisites

- An Azure account with an active subscription. [Create an account for free](#).
- An Azure Virtual Machine with a public IP address assigned to its network interface. For more information on creating a virtual machine with a public IP, see [Quickstart: Create a Windows virtual machine in the Azure portal](#).
  - For the purposes of this article, the example virtual machine is named **myVM**. The example public IP address is named **myPublicIP**.

### NOTE

Removal of the public IP address prevents direct connections to the virtual machine from the internet. RDP or SSH access won't function to the virtual machine after you complete this migration. To securely manage virtual machines in your subscription, use Azure Bastion. For more information on Azure Bastion, see [What is Azure Bastion?](#)

## Remove public IP from virtual machine

In this section, you'll learn how to remove the public IP address from the virtual machine.

1. Sign in to the [Azure portal](#).
2. In the search box at the top of the portal, enter **Virtual machine**. Select **Virtual machines**.
3. In **Virtual machines**, select **myVM** or your virtual machine.
4. In the **Overview** of **myVM**, select **Public IP address**.

Home > myResourceGroup >

myVM Virtual machine

Search (Ctrl+ /) Connect Start Restart Stop Capture Delete Refresh

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

Settings Networking Connect Windows Admin Center (preview)

Essentials

- Resource group ([move](#)) [myResourceGroup](#)
- Status Running
- Location West US 2
- Subscription ([move](#)) [Contoso Subscription](#)
- Subscription ID
- Tags ([edit](#)) [Click here to add tags](#)

Operating system Windows (Windows Server 2019 Datacenter)

Size Standard B2s (2 vcpus, 4 GiB memory)

Public IP address **20.112.99.221**

Virtual network/subnet [myResourceGroup-vnet/default](#)

DNS name [Not configured](#)

JSON View

5. In **myPublicIP**, select the **Overview** page in the left-hand column.

6. In **Overview**, select **Dissociate**.

Home > myResourceGroup > myVM >

myPublicIP Public IP address

Search (Ctrl+ /) Associate Dissociate Move Delete Refresh

Overview Activity log Access control (IAM) Tags

Settings Configuration Properties Locks

Monitoring Insights Alerts Metrics

Essentials

Upgrade to Standard SKU - Microsoft recommends Standard SKU public IP address for production workloads

Resource group ( <a href="#">move</a> )	: <a href="#">myResourceGroup</a>
Location	: West US 2
Subscription ( <a href="#">move</a> )	: <a href="#">Contoso Subscription</a>
Subscription ID	:
SKU	: Basic
Tier	: Regional
IP address	: 20.112.99.221
DNS name	: -
Associated to	: <a href="#">myvm912</a>
Tags ( <a href="#">edit</a> )	: <a href="#">Click here to add tags</a>
See more	

JSON View

7. Select **Yes** in **Dissociate public IP address**.

#### (Optional) Upgrade IP address

The NAT gateway resource in Azure Virtual Network NAT requires a standard SKU public IP address. In this section, you'll upgrade the IP you removed from the virtual machine in the previous section. If the IP address you removed is already a standard SKU public IP, you can proceed to the next section.

1. In the search box at the top of the portal, enter **Public IP**. Select **Public IP addresses**.
2. In **Public IP addresses**, select **myPublicIP** or your basic SKU IP address.
3. In the **Overview** of **myPublicIP**, select the IP address upgrade banner.

myPublicIP

Public IP address

Associate Dissociate Move Delete Refresh

Overview Activity log Access control (IAM) Tags

Upgrades to Standard SKU - Microsoft recommends Standard SKU public IP address for production workloads

Essentials Resource group (move) : myResourceGroup JSON View

- In Upgrade to Standard SKU, select the box next to I acknowledge. Select the Upgrade button.

myPublicIP

Public IP address

Associate Dissociate Move Delete Refresh

Overview Activity log Access control (IAM) Tags Settings

Upgrade to Standard SKU

You won't be able to revert the Standard Public IP address back to Basic SKU once you upgrade it. There is a cost associated with Standard Public IP address whereas Basic Public IP address is free. For comparison and pricing, [Learn more](#).

I acknowledge.

Upgrade Cancel

- When the upgrade is complete, proceed to the next section.

## Create NAT gateway

In this section, you'll create a NAT gateway with the IP address you previously removed from the virtual machine. You'll assign the NAT gateway to your pre-created subnet within your virtual network. The subnet name for this example is **default**.

- In the search box at the top of the portal, enter **NAT gateway**. Select **NAT gateways**.
- In **NAT gateways**, select **+ Create**.
- In **Create network address translation (NAT) gateway**, enter or select the following information in the **Basics** tab.

SETTING	VALUE
<b>Project details</b>	
Subscription	Select your subscription.
Resource group	Select <b>Create new</b> . Enter <b>myResourceGroup</b> . Select <b>OK</b> .
<b>Instance details</b>	
NAT gateway name	Enter <b>myNATgateway</b> .
Region	Select the region of your virtual network. In this example, it's <b>West US 2</b> .

SETTING	VALUE
Availability zone	Leave the default of <b>None</b> .
Idle timeout (minutes)	Enter <b>10</b> .

4. Select the **Outbound IP** tab, or select **Next: Outbound IP** at the bottom of the page.
5. In **Public IP addresses** in the **Outbound IP** tab, select the IP address from the previous section in **Public IP addresses**. In this example, it's **myPublicIP**.
6. Select the **Subnet** tab, or select **Next: Subnet** at the bottom of the page.
7. In the pull-down box for **Virtual network**, select your virtual network.
8. In **Subnet name**, select the checkbox for your subnet. In this example, it's **default**.
9. Select the **Review + create** tab, or select **Review + create** at the bottom of the page.
10. Select **Create**.

## Clean up resources

If you're not going to continue to use this application, delete the NAT gateway with the following steps:

1. From the left-hand menu, select **Resource groups**.
2. Select the **myResourceGroup** resource group.
3. Select **Delete resource group**.
4. Enter **myResourceGroup** and select **Delete**.

## Next steps

In this article, you learned how to:

- Remove a public IP address from a virtual machine.
- Create a NAT gateway and use the public IP address from the virtual machine for the NAT gateway resource.

Any virtual machine created within this subnet won't require a public IP address and will automatically have outbound connectivity. For more information about NAT gateway and the connectivity benefits it provides, see [Design virtual networks with NAT gateway](#).

Advance to the next article to learn how to migrate default outbound access to Azure Virtual Network NAT:

[Migrate outbound access to NAT gateway](#)

# Create a virtual machine with a static public IP address using the Azure CLI

9/21/2022 • 3 minutes to read • [Edit Online](#)

In this article, you'll create a VM with a static public IP address. A public IP address enables communication to a virtual machine from the internet. Assign a static public IP address, instead of a dynamic address, to ensure the address never changes.

Public IP addresses have a [nominal charge](#). There's a [limit](#) to the number of public IP addresses that you can use per subscription.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).
- An Azure account with an active subscription. [Create an account for free](#).
- This tutorial requires version 2.0.28 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a resource group

An Azure resource group is a logical container into which Azure resources are deployed and managed.

Create a resource group with [az group create](#) named **myResourceGroup** in the **eastus2** location.

```
az group create \
--name myResourceGroup \
--location eastus2
```

## Create a public IP address

Use [az network public-ip create](#) to create a standard public IPv4 address.

The following command creates a zone-redundant public IP address named **myPublicIP** in

myResourceGroup.

```
az network public-ip create \
--resource-group myResourceGroup \
--name myPublicIP \
--version IPv4 \
--sku Standard \
--zone 1 2 3
```

## Create a virtual machine

Create a virtual machine with [az vm create](#).

The following command creates a Windows Server virtual machine. You'll enter the name of the public IP address created previously in the `-PublicIPAddressName` parameter. When prompted, provide a username and password to be used as the credentials for the virtual machine:

```
az vm create \
--name myVM \
--resource-group TutorVMRoutePref-rg \
--public-ip-address myPublicIP \
--size Standard_A2 \
--image MicrosoftWindowsServer:WindowsServer:2019-Datacenter:latest \
--admin-username azureuser
```

For more information on public IP SKUs, see [Public IP address SKUs](#). A virtual machine can be added to the backend pool of an Azure Load Balancer. The SKU of the public IP address must match the SKU of a load balancer's public IP. For more information, see [Azure Load Balancer](#).

View the public IP address assigned and confirm that it was created as a static address, with [az network public-ip show](#):

```
az network public-ip show \
--resource-group myResourceGroup \
--name myPublicIP \
--query [ipAddress,publicIpAllocationMethod,sku] \
--output table
```

### WARNING

Do not modify the IP address settings within the virtual machine's operating system. The operating system is unaware of Azure public IP addresses. Though you can add private IP address settings to the operating system, we recommend not doing so unless necessary, and not until after reading [Add a private IP address to an operating system](#).

#### NOTE

Azure provides a default outbound access IP for VMs that either aren't assigned a public IP address or are in the back-end pool of an internal basic Azure load balancer. The default outbound access IP mechanism provides an outbound IP address that isn't configurable.

For more information, see [Default outbound access in Azure](#).

The default outbound access IP is disabled when either a public IP address is assigned to the VM or the VM is placed in the back-end pool of a standard load balancer, with or without outbound rules. If an [Azure Virtual Network network address translation \(NAT\) gateway](#) resource is assigned to the subnet of the virtual machine, the default outbound access IP is disabled.

VMs that are created by virtual machine scale sets in flexible orchestration mode don't have default outbound access.

For more information about outbound connections in Azure, see [Use source network address translation \(SNAT\) for outbound connections](#).

## Clean up resources

When no longer needed, you can use `az group delete` to remove the resource group and all of the resources it contains:

```
az group delete --name myResourceGroup --yes
```

## Next steps

- Learn more about [public IP addresses](#) in Azure.
- Learn more about all [public IP address settings](#).
- Learn more about [private IP addresses](#) and assigning a [static private IP address](#) to an Azure virtual machine.
- Learn more about creating [Linux](#) and [Windows](#) virtual machines.

# Add Custom Domain to Azure VM or resource

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

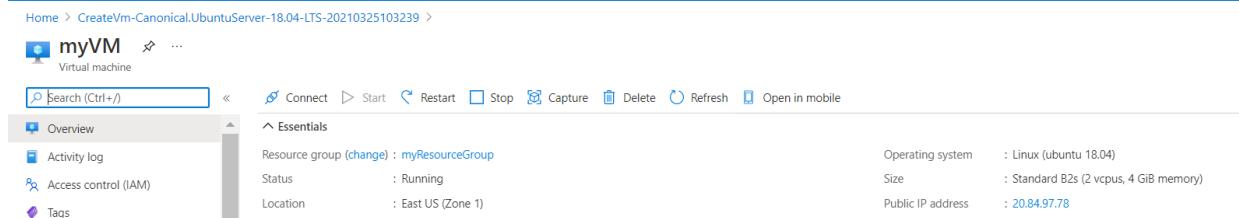
In Azure there are multiple ways to connect a custom domain to your VM or resource. For any resource with a public IP (Virtual Machine, Load Balancer, Application Gateway) the most straight-forward way is to create an A record set in your corresponding domain registrar.

## Prerequisites

- You need a VM with a web server running. You can use the [Quickstart](#) to create a VM and add NGINX.
- The VM must be accessible to the web (open port 80, or 443). For a more secure deployment place your VM behind a load balancer or Application Gateway first. For more information, see [Quickstart: Load Balancer](#).
- Have an existing domain and access to DNS settings. For more information, see [Buy a custom domain for Azure App Service](#).

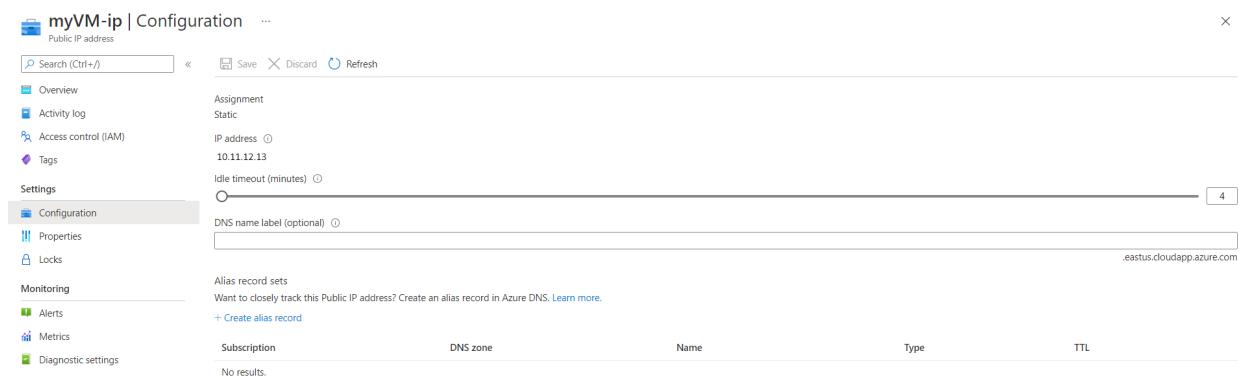
## Add custom domain to VM public IP address

When you create a virtual machine in the Azure portal, a public IP resource for the virtual machine is automatically created. Your public IP address is shown in VM overview page.



The screenshot shows the Azure portal interface for a virtual machine named 'myVM'. In the top navigation bar, the URL is 'Home > CreateVm-Canonical.UbuntuServer-18.04-LTS-20210325103239 >'. Below the navigation bar, there's a search bar and a toolbar with options like 'Connect', 'Start', 'Stop', 'Capture', 'Delete', 'Refresh', and 'Open in mobile'. On the left, a sidebar menu includes 'Overview', 'Activity log', 'Access control (IAM)', and 'Tags'. The main content area is titled 'Essentials' and displays the following details:  
Resource group (change) : myResourceGroup  
Status : Running  
Location : East US (Zone 1)  
Operating system : Linux (ubuntu 18.04)  
Size : Standard B2s (2 vcpus, 4 GiB memory)  
Public IP address : 20.84.97.78

If you select the IP address you can see more information on it. Check to make sure your **IP Assignment** is set to **Static**. A static IP address will not change if the VM or resource reboots or shuts down.



The screenshot shows the 'Configuration' page for the public IP address 'myVM-ip'. The left sidebar includes 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Settings' (with 'Configuration' selected), 'Monitoring', 'Alerts', 'Metrics', and 'Diagnostic settings'. The main content area has tabs for 'Configuration' and 'Properties'. Under 'Configuration', there are fields for 'Assignment' (set to 'Static'), 'IP address' (10.11.12.13), 'Idle timeout (minutes)' (set to 4), and 'DNS name label (optional)' (eastus.cloudapp.azure.com). Below these fields, there's a note about alias record sets and a link to create one. At the bottom, there's a table for 'Subscription', 'DNS zone', 'Name', 'Type', and 'TTL', with a note 'No results.'

If your IP Address is not static, you will need to create an FQDN.

1. Select your VM in the portal.
2. In the left menu, select **Properties**
3. Under **Public IP address\DNS name label**, select your IP address.
4. Under **DNS name label**, enter the prefix you want to use.

5. Select **Save** at the top of the page.
6. Select **Overview** in the left menu to return to the VM overview blade.
7. Verify that the *DNS name* appears correctly.

Open a browser and enter your IP address or FQDN and verify that it shows the web content running on your VM.

After verifying your static IP or FQDN, go to your domain provider and navigate to DNS settings.

Once there add an *A record* pointing to your Public IP Address or FQDN. For example, the procedure for the GoDaddy domain registrar is as follows:

1. Sign in and select the custom domain you want to use.
2. In the **Domains** section, select **Manage All**, then select **DNS | Manage Zones**.
3. For **Domain Name**, enter your custom domain, then select **Search**.
4. From the DNS Management page, select **Add**, then select **A** in the Type list.
5. Complete the fields of the A entry:
  - Type: Leave **A** selected.
  - Host: Enter **@**
  - Points to: Enter the Public IP Address or FQDN of your VM.
  - TTL: Leave one hour selected.
6. Select **Save**.

The A record entry is added to the DNS records table.

After the record is created it usually takes about an hour for DNS propagate, but it can sometimes take up to 48 hours.

## Next steps

[Overview of TLS termination and end to end TLS with Application Gateway.](#)

# How to create a Linux virtual machine in Azure with multiple network interface cards

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article details how to create a VM with multiple NICs with the Azure CLI.

## Create supporting resources

Install the latest [Azure CLI](#) and log in to an Azure account using [az login](#).

In the following examples, replace example parameter names with your own values. Example parameter names included *myResourceGroup*, *mystorageaccount*, and *myVM*.

First, create a resource group with [az group create](#). The following example creates a resource group named *myResourceGroup* in the *eastus* location:

```
az group create --name myResourceGroup --location eastus
```

Create the virtual network with [az network vnet create](#). The following example creates a virtual network named *myVnet* and subnet named *mySubnetFrontEnd*.

```
az network vnet create \
--resource-group myResourceGroup \
--name myVnet \
--address-prefix 10.0.0.0/16 \
--subnet-name mySubnetFrontEnd \
--subnet-prefix 10.0.1.0/24
```

Create a subnet for the back-end traffic with [az network vnet subnet create](#). The following example creates a subnet named *mySubnetBackEnd*.

```
az network vnet subnet create \
--resource-group myResourceGroup \
--vnet-name myVnet \
--name mySubnetBackEnd \
--address-prefix 10.0.2.0/24
```

Create a network security group with [az network nsg create](#). The following example creates a network security group named *myNetworkSecurityGroup*.

```
az network nsg create \
--resource-group myResourceGroup \
--name myNetworkSecurityGroup
```

## Create and configure multiple NICs

Create two NICs with [az network nic create](#). The following example creates two NICs, named *myNic1* and *myNic2*, connected the network security group, with one NIC connecting to each subnet:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic1 \
--vnet-name myVnet \
--subnet mySubnetFrontEnd \
--network-security-group myNetworkSecurityGroup
az network nic create \
--resource-group myResourceGroup \
--name myNic2 \
--vnet-name myVnet \
--subnet mySubnetBackEnd \
--network-security-group myNetworkSecurityGroup
```

## Create a VM and attach the NICs

When you create the VM, specify the NICs you created with `--nics`. You also need to take care when you select the VM size. There are limits for the total number of NICs that you can add to a VM. Read more about [Linux VM sizes](#).

Create a VM with [az vm create](#). The following example creates a VM named *myVM*:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--image UbuntuLTS \
--size Standard_DS3_v2 \
--admin-username azureuser \
--generate-ssh-keys \
--nics myNic1 myNic2
```

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Add a NIC to a VM

The previous steps created a VM with multiple NICs. You can also add NICs to an existing VM with the Azure CLI. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly. If needed, you can [resize a VM](#).

Create another NIC with [az network nic create](#). The following example creates a NIC named *myNic3* connected to the back-end subnet and network security group created in the previous steps:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic3 \
--vnet-name myVnet \
--subnet mySubnetBackEnd \
--network-security-group myNetworkSecurityGroup
```

To add a NIC to an existing VM, first deallocate the VM with [az vm deallocate](#). The following example deallocates the VM named *myVM*:

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

Add the NIC with [az vm nic add](#). The following example adds *myNic3* to *myVM*:

```
az vm nic add \
--resource-group myResourceGroup \
--vm-name myVM \
--nics myNic3
```

Start the VM with [az vm start](#):

```
az vm start --resource-group myResourceGroup --name myVM
```

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Remove a NIC from a VM

To remove a NIC from an existing VM, first deallocate the VM with [az vm deallocate](#). The following example deallocates the VM named *myVM*.

```
az vm deallocate --resource-group myResourceGroup --name myVM
```

Remove the NIC with [az vm nic remove](#). The following example removes *myNic3* from *myVM*.

```
az vm nic remove \
--resource-group myResourceGroup \
--vm-name myVM \
--nics myNic3
```

Start the VM with [az vm start](#):

```
az vm start --resource-group myResourceGroup --name myVM
```

## Create multiple NICs using Resource Manager templates

Azure Resource Manager templates use declarative JSON files to define your environment. You can read an [overview of Azure Resource Manager](#). Resource Manager templates provide a way to create multiple instances of a resource during deployment, such as creating multiple NICs. You use *copy* to specify the number of instances to create:

```
"copy": {
  "name": "multiplenics"
  "count": "[parameters('count')]"
}
```

Read more about [creating multiple instances using copy](#).

You can also use a `copyIndex()` to then append a number to a resource name, which allows you to create `myNic1`, `myNic2`, etc. The following shows an example of appending the index value:

```
"name": "[concat('myNic', copyIndex())]",
```

You can read a complete example of [creating multiple NICs using Resource Manager templates](#).

Add routing tables to the guest OS by completing the steps in [Configure the guest OS for multiple NICs](#).

## Configure guest OS for multiple NICs

The previous steps created a virtual network and subnet, attached NICs, then created a VM. A public IP address and network security group rules that allow SSH traffic were not created. To configure the guest OS for multiple NICs, you need to allow remote connections and run commands locally on the VM.

To allow SSH traffic, create a network security group rule with [az network nsg rule create](#) as follows:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name allow_ssh \
--priority 101 \
--destination-port-ranges 22
```

Create a public IP address with [az network public-ip create](#) and assign it to the first NIC with [az network nic ip-config update](#):

```
az network public-ip create --resource-group myResourceGroup --name myPublicIP

az network nic ip-config update \
--resource-group myResourceGroup \
--nic-name myNic1 \
--name ipconfig1 \
--public-ip myPublicIP
```

To view the public IP address of the VM, use [az vm show](#) as follows::

```
az vm show --resource-group myResourceGroup --name myVM -d --query publicIps -o tsv
```

Now SSH to the public IP address of your VM. The default username provided in a previous step was *azureuser*. Provide your own username and public IP address:

```
ssh azureuser@137.117.58.232
```

To send to or from a secondary network interface, you have to manually add persistent routes to the operating system for each secondary network interface. In this article, *eth1* is the secondary interface. Instructions for adding persistent routes to the operating system vary by distro. See documentation for your distro for instructions.

When adding the route to the operating system, the gateway address is the first address of the subnet the network interface is in. For example, if the subnet has been assigned the range *10.0.2.0/24*, the gateway you specify for the route is *10.0.2.1* or if the subnet has been assigned the range *10.0.2.128/25*, the gateway you specify for the route is *10.0.2.129*. You can define a specific network for the route's destination, or specify a destination of *0.0.0.0*, if you want all traffic for the interface to go through the specified gateway. The gateway for each subnet is managed by the virtual network.

Once you've added the route for a secondary interface, verify that the route is in your route table with `route -n`. The following example output is for the route table that has the two network interfaces added to the VM in this article:

Kernel IP routing table							
Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
0.0.0.0	10.0.1.1	0.0.0.0	UG	0	0	0	eth0
0.0.0.0	10.0.2.1	0.0.0.0	UG	0	0	0	eth1
10.0.1.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
10.0.2.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
168.63.129.16	10.0.1.1	255.255.255.255	UGH	0	0	0	eth0
169.254.169.254	10.0.1.1	255.255.255.255	UGH	0	0	0	eth0

Confirm that the route you added persists across reboots by checking your route table again after a reboot. To test connectivity, you can enter the following command, for example, where *eth1* is the name of a secondary network interface:

```
ping bing.com -c 4 -I eth1
```

## Next steps

Review [Linux VM sizes](#) when trying to creating a VM with multiple NICs. Pay attention to the maximum number of NICs each VM size supports.

To further secure your VMs, use just in time VM access. This feature opens network security group rules for SSH traffic when needed, and for a defined period of time. For more information, see [Manage virtual machine access using just in time](#).

# Create and manage a Windows virtual machine that has multiple NICs

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

Virtual machines (VMs) in Azure can have multiple virtual network interface cards (NICs) attached to them. A common scenario is to have different subnets for front-end and back-end connectivity. You can associate multiple NICs on a VM to multiple subnets, but those subnets must all reside in the same virtual network (vNet). This article details how to create a VM that has multiple NICs attached to it. You also learn how to add or remove NICs from an existing VM. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly.

## NOTE

If multiple subnets are not required for a scenario, it may be more straightforward to utilize multiple IP configurations on a single NIC. Instructions for this setup can be found [here](#).

## Prerequisites

In the following examples, replace example parameter names with your own values. Example parameter names include *myResourceGroup*, *myVnet*, and *myVM*.

## Create a VM with multiple NICs

First, create a resource group. The following example creates a resource group named *myResourceGroup* in the *EastUs* location:

```
New-AzResourceGroup -Name "myResourceGroup" -Location "EastUS"
```

### Create virtual network and subnets

A common scenario is for a virtual network to have two or more subnets. One subnet may be for front-end traffic, the other for back-end traffic. To connect to both subnets, you then use multiple NICs on your VM.

1. Define two virtual network subnets with [New-AzVirtualNetworkSubnetConfig](#). The following example defines the subnets for *mySubnetFrontEnd* and *mySubnetBackEnd*.

```
$mySubnetFrontEnd = New-AzVirtualNetworkSubnetConfig -Name "mySubnetFrontEnd" `  
    -AddressPrefix "192.168.1.0/24"  
$mySubnetBackEnd = New-AzVirtualNetworkSubnetConfig -Name "mySubnetBackEnd" `  
    -AddressPrefix "192.168.2.0/24"
```

2. Create your virtual network and subnets with [New-AzVirtualNetwork](#). The following example creates a virtual network named *myVnet*.

```
$myVnet = New-AzVirtualNetwork -ResourceGroupName "myResourceGroup" `  
    -Location "EastUs" `  
    -Name "myVnet" `  
    -AddressPrefix "192.168.0.0/16" `  
    -Subnet $mySubnetFrontEnd,$mySubnetBackEnd
```

## Create multiple NICs

Create two NICs with [New-AzNetworkInterface](#). Attach one NIC to the front-end subnet and one NIC to the back-end subnet. The following example creates NICs named *myNic1* and *myNic2*:

```
$frontEnd = $myVnet.Subnets | ?{$_ . Name -eq 'mySubnetFrontEnd'}  
$myNic1 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" `  
    -Name "myNic1" `  
    -Location "EastUs" `  
    -SubnetId $frontEnd.Id  
  
$backEnd = $myVnet.Subnets | ?{$_ . Name -eq 'mySubnetBackEnd'}  
$myNic2 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" `  
    -Name "myNic2" `  
    -Location "EastUs" `  
    -SubnetId $backEnd.Id
```

Typically you also create a [network security group](#) to filter network traffic to the VM and a [load balancer](#) to distribute traffic across multiple VMs.

## Create the virtual machine

Now start to build your VM configuration. Each VM size has a limit for the total number of NICs that you can add to a VM. For more information, see [Windows VM sizes](#).

1. Set your VM credentials to the `$cred` variable as follows:

```
$cred = Get-Credential
```

2. Define your VM with [New-AzVMConfig](#). The following example defines a VM named *myVM* and uses a VM size that supports more than two NICs (*Standard\_DS3\_v2*):

```
$vmConfig = New-AzVMConfig -VMName "myVM" -VMSize "Standard_DS3_v2"
```

3. Create the rest of your VM configuration with [Set-AzVMOperatingSystem](#) and [Set-AzVMSourceImage](#). The following example creates a Windows Server 2016 VM:

```
$vmConfig = Set-AzVMOperatingSystem -VM $vmConfig `  
    -Windows `  
    -ComputerName "myVM" `  
    -Credential $cred `  
    -ProvisionVMAgent `  
    -EnableAutoUpdate  
$vmConfig = Set-AzVMSourceImage -VM $vmConfig `  
    -PublisherName "MicrosoftWindowsServer" `  
    -Offer "WindowsServer" `  
    -Skus "2016-Datacenter" `  
    -Version "latest"
```

4. Attach the two NICs that you previously created with [Add-AzVMNetworkInterface](#):

```
$vmConfig = Add-AzVMNetworkInterface -VM $vmConfig -Id $myNic1.Id -Primary  
$vmConfig = Add-AzVMNetworkInterface -VM $vmConfig -Id $myNic2.Id
```

5. Create your VM with [New-AzVM](#):

```
New-AzVM -VM $vmConfig -ResourceGroupName "myResourceGroup" -Location "EastUs"
```

6. Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Add a NIC to an existing VM

To add a virtual NIC to an existing VM, you deallocate the VM, add the virtual NIC, then start the VM. Different [VM sizes](#) support a varying number of NICs, so size your VM accordingly. If needed, you can [resize a VM](#).

1. Deallocate the VM with [Stop-AzVM](#). The following example deallocates the VM named *myVM* in *myResourceGroup*:

```
Stop-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

2. Get the existing configuration of the VM with [Get-AzVm](#). The following example gets information for the VM named *myVM* in *myResourceGroup*:

```
$vm = Get-AzVm -Name "myVM" -ResourceGroupName "myResourceGroup"
```

3. The following example creates a virtual NIC with [New-AzNetworkInterface](#) named *myNic3* that is attached to *mySubnetBackEnd*. The virtual NIC is then attached to the VM named *myVM* in *myResourceGroup* with [Add-AzVMNetworkInterface](#):

```
# Get info for the back end subnet  
$myVnet = Get-AzVirtualNetwork -Name "myVnet" -ResourceGroupName "myResourceGroup"  
$backEnd = $myVnet.Subnets | ?{$_ . Name -eq 'mySubnetBackEnd' }  
  
# Create a virtual NIC  
$myNic3 = New-AzNetworkInterface -ResourceGroupName "myResourceGroup" `  
    -Name "myNic3" `  
    -Location "EastUs" `  
    -SubnetId $backEnd.Id  
  
# Get the ID of the new virtual NIC and add to VM  
$nicId = (Get-AzNetworkInterface -ResourceGroupName "myResourceGroup" -Name "MyNic3").Id  
Add-AzVMNetworkInterface -VM $vm -Id $nicId | Update-AzVm -ResourceGroupName "myResourceGroup"
```

### Primary virtual NICs

One of the NICs on a multi-NIC VM needs to be primary. If one of the existing virtual NICs on the VM is already set as primary, you can skip this step. The following example assumes that two virtual NICs are now present on a VM and you wish to add the first NIC ([0]) as the primary:

```
# List existing NICs on the VM and find which one is primary  
$vm.NetworkProfile.NetworkInterfaces  
  
# Set NIC 0 to be primary  
$vm.NetworkProfile.NetworkInterfaces[0].Primary = $true  
$vm.NetworkProfile.NetworkInterfaces[1].Primary = $false  
  
# Update the VM state in Azure  
Update-AzVM -VM $vm -ResourceGroupName "myResourceGroup"
```

#### 4. Start the VM with [Start-AzVm](#):

```
Start-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM"
```

#### 5. Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Remove a NIC from an existing VM

To remove a virtual NIC from an existing VM, you deallocate the VM, remove the virtual NIC, then start the VM.

#### 1. Deallocation the VM with [Stop-AzVM](#). The following example deallocated the VM named *myVM* in *myResourceGroup*:

```
Stop-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

#### 2. Get the existing configuration of the VM with [Get-AzVm](#). The following example gets information for the VM named *myVM* in *myResourceGroup*.

```
$vm = Get-AzVm -Name "myVM" -ResourceGroupName "myResourceGroup"
```

#### 3. Get information about the NIC remove with [Get-AzNetworkInterface](#). The following example gets information about *myNic3*:

```
# List existing NICs on the VM if you need to determine NIC name  
$vm.NetworkProfile.NetworkInterfaces  
  
$nicId = (Get-AzNetworkInterface -ResourceGroupName "myResourceGroup" -Name "myNic3").Id
```

#### 4. Remove the NIC with [Remove-AzVMNetworkInterface](#) and then update the VM with [Update-AzVm](#). The following example removes *myNic3* as obtained by `$nicId` in the preceding step:

```
Remove-AzVMNetworkInterface -VM $vm -NetworkInterfaceIDs $nicId | `  
Update-AzVm -ResourceGroupName "myResourceGroup"
```

#### 5. Start the VM with [Start-AzVm](#):

```
Start-AzVM -Name "myVM" -ResourceGroupName "myResourceGroup"
```

## Create multiple NICs with templates

Azure Resource Manager templates provide a way to create multiple instances of a resource during deployment,

such as creating multiple NICs. Resource Manager templates use declarative JSON files to define your environment. For more information, see [overview of Azure Resource Manager](#). You can use `copy` to specify the number of instances to create:

```
"copy": {  
    "name": "multiplenics",  
    "count": "[parameters('count')]"  
}
```

For more information, see [creating multiple instances by using `copy`](#).

You can also use `copyIndex()` to append a number to a resource name. You can then create `myNic1`, `MyNic2` and so on. The following code shows an example of appending the index value:

```
"name": "[concat('myNic', copyIndex())]",
```

You can read a complete example of [creating multiple NICs by using Resource Manager templates](#).

Add routes for secondary NICs to the OS by completing the steps in [Configure the operating system for multiple NICs](#).

## Configure guest OS for multiple NICs

Azure assigns a default gateway to the first (primary) network interface attached to the virtual machine. Azure does not assign a default gateway to additional (secondary) network interfaces attached to a virtual machine. Therefore, you are unable to communicate with resources outside the subnet that a secondary network interface is in, by default. Secondary network interfaces can, however, communicate with resources outside their subnet, though the steps to enable communication are different for different operating systems.

- From a Windows command prompt, run the `route print` command, which returns output similar to the following output for a virtual machine with two attached network interfaces:

```
=====  
Interface List  
3...00 0d 3a 10 92 ce .....Microsoft Hyper-V Network Adapter #3  
7...00 0d 3a 10 9b 2a .....Microsoft Hyper-V Network Adapter #4  
=====
```

In this example, **Microsoft Hyper-V Network Adapter #4** (interface 7) is the secondary network interface that doesn't have a default gateway assigned to it.

- From a command prompt, run the `ipconfig` command to see which IP address is assigned to the secondary network interface. In this example, 192.168.2.4 is assigned to interface 7. No default gateway address is returned for the secondary network interface.
- To route all traffic destined for addresses outside the subnet of the secondary network interface to the gateway for the subnet, run the following command:

```
route add -p 0.0.0.0 MASK 0.0.0.0 192.168.2.1 METRIC 5015 IF 7
```

The gateway address for the subnet is the first IP address (ending in .1) in the address range defined for the subnet. If you don't want to route all traffic outside the subnet, you could add individual routes to specific destinations, instead. For example, if you only wanted to route traffic from the secondary network interface to the 192.168.3.0 network, you enter the command:

```
route add -p 192.168.3.0 MASK 255.255.255.0 192.168.2.1 METRIC 5015 IF 7
```

4. To confirm successful communication with a resource on the 192.168.3.0 network, for example, enter the following command to ping 192.168.3.4 using interface 7 (192.168.2.4):

```
ping 192.168.3.4 -S 192.168.2.4
```

You may need to open ICMP through the Windows firewall of the device you're pinging with the following command:

```
netsh advfirewall firewall add rule name=Allow-ping protocol=icmpv4 dir=in action=allow
```

5. To confirm the added route is in the route table, enter the `route print` command, which returns output similar to the following text:

```
=====
Active Routes:
Network Destination      Netmask          Gateway        Interface Metric
      0.0.0.0          0.0.0.0    192.168.1.1    192.168.1.4      15
      0.0.0.0          0.0.0.0    192.168.2.1    192.168.2.4    5015
```

The route listed with **192.168.1.1** under **Gateway**, is the route that is there by default for the primary network interface. The route with **192.168.2.1** under **Gateway**, is the route you added.

## Next steps

Review [Windows VM sizes](#) when you're trying to create a VM that has multiple NICs. Pay attention to the maximum number of NICs that each VM size supports.

# Create a fully qualified domain name for a VM in the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: Applies to: ✓ Linux VMs ✓ Windows VMs

When you create a virtual machine (VM) in the [Azure portal](#), a public IP resource for the virtual machine is automatically created. You use this public IP address to remotely access the VM. Although the portal does not create a [fully qualified domain name](#), or FQDN, you can add one once the VM is created. This article demonstrates the steps to create a DNS name or FQDN. If you create a VM without a public IP address, you can't create a FQDN.

## Create a FQDN

This article assumes that you have already created a VM. If needed, you can create a [Linux](#) or [Windows](#) VM in the portal. Follow these steps once your VM is up and running:

1. Select your VM in the portal.
2. In the left menu, select **Properties**
3. Under **Public IP address\DNS name label**, select your IP address.
4. Under **DNS name label**, enter the prefix you want to use.
5. Select **Save** at the top of the page.
6. Select **Overview** in the left menu to return to the VM overview blade.
7. Verify that the **DNS name** appears correctly.

## Next steps

You can also manage DNS using [Azure DNS zones](#).

# Create a fully qualified domain name for a VM in the Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: Applies to: ✓ Linux VMs ✓ Windows VMs

When you create a virtual machine (VM) in the [Azure portal](#), a public IP resource for the virtual machine is automatically created. You use this public IP address to remotely access the VM. Although the portal does not create a [fully qualified domain name](#), or FQDN, you can add one once the VM is created. This article demonstrates the steps to create a DNS name or FQDN. If you create a VM without a public IP address, you can't create a FQDN.

## Create a FQDN

This article assumes that you have already created a VM. If needed, you can create a [Linux](#) or [Windows](#) VM in the portal. Follow these steps once your VM is up and running:

1. Select your VM in the portal.
2. In the left menu, select **Properties**
3. Under **Public IP address\DNS name label**, select your IP address.
4. Under **DNS name label**, enter the prefix you want to use.
5. Select **Save** at the top of the page.
6. Select **Overview** in the left menu to return to the VM overview blade.
7. Verify that the **DNS name** appears correctly.

## Next steps

You can also manage DNS using [Azure DNS zones](#).

# DNS Name Resolution options for Linux virtual machines in Azure

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure provides DNS name resolution by default for all virtual machines that are in a single virtual network. You can implement your own DNS name resolution solution by configuring your own DNS services on your virtual machines that Azure hosts. The following scenarios should help you choose the one that works for your situation.

- [Name resolution that Azure provides](#)
- [Name resolution using your own DNS server](#)

The type of name resolution that you use depends on how your virtual machines and role instances need to communicate with each other.

The following table illustrates scenarios and corresponding name resolution solutions:

SCENARIO	SOLUTION	SUFFIX
Name resolution between role instances or virtual machines in the same virtual network	Name resolution that Azure provides	hostname or fully-qualified domain name (FQDN)
Name resolution between role instances or virtual machines in different virtual networks	Customer-managed DNS servers that forward queries between virtual networks for resolution by Azure (DNS proxy). See <a href="#">Name resolution using your own DNS server</a> .	FQDN only
Resolution of on-premises computers and service names from role instances or virtual machines in Azure	Customer-managed DNS servers (for example, on-premises domain controller, local read-only domain controller, or a DNS secondary synced by using zone transfers). See <a href="#">Name resolution using your own DNS server</a> .	FQDN only
Resolution of Azure hostnames from on-premises computers	Forward queries to a customer-managed DNS proxy server in the corresponding virtual network. The proxy server forwards queries to Azure for resolution. See <a href="#">Name resolution using your own DNS server</a> .	FQDN only
Reverse DNS for internal IPs	<a href="#">Name resolution using your own DNS server</a>	n/a

## Name resolution that Azure provides

Along with resolution of public DNS names, Azure provides internal name resolution for virtual machines and role instances that are in the same virtual network. In virtual networks that are based on Azure Resource Manager, the DNS suffix is consistent across the virtual network; the FQDN is not needed. DNS names can be

assigned to both network interface cards (NICs) and virtual machines. Although the name resolution that Azure provides does not require any configuration, it is not the appropriate choice for all deployment scenarios, as seen on the preceding table.

## Features and considerations

### Features:

- No configuration is required to use name resolution that Azure provides.
- The name resolution service that Azure provides is highly available. You don't need to create and manage clusters of your own DNS servers.
- The name resolution service that Azure provides can be used along with your own DNS servers to resolve both on-premises and Azure hostnames.
- Name resolution is provided between virtual machines in virtual networks without need for the FQDN.
- You can use hostnames that best describe your deployments rather than working with auto-generated names.

### Considerations:

- The DNS suffix that Azure creates cannot be modified.
- You cannot manually register your own records.
- WINS and NetBIOS are not supported.
- Hostnames must be DNS-compatible. Names must use only 0-9, a-z, and '-', and they cannot start or end with a '-'. See RFC 3696 Section 2.
- DNS query traffic is throttled for each virtual machine. Throttling shouldn't impact most applications. If request throttling is observed, ensure that client-side caching is enabled. For more information, see [Getting the most from name resolution that Azure provides](#).

## Getting the most from name resolution that Azure provides

### Client-side caching:

Some DNS queries are not sent across the network. Client-side caching helps reduce latency and improve resilience to network inconsistencies by resolving recurring DNS queries from a local cache. DNS records contain a Time-To-Live (TTL), which enables the cache to store the record for as long as possible without impacting record freshness. As a result, client-side caching is suitable for most situations.

Some Linux distributions do not include caching by default. We recommend that you add a cache to each Linux virtual machine after you check that there isn't a local cache already.

Several different DNS caching packages, such as dnsmasq, are available. Here are the steps to install dnsmasq on the most common distributions:

### Ubuntu (uses resolvconf)

- Install the dnsmasq package ("sudo apt-get install dnsmasq").

### SUSE (uses netconf):

1. Install the dnsmasq package ("sudo zypper install dnsmasq").
2. Enable the dnsmasq service ("systemctl enable dnsmasq.service").
3. Start the dnsmasq service ("systemctl start dnsmasq.service").
4. Edit "/etc/sysconfig/network/config", and change NETCONFIG\_DNS\_FORWARDER="" to "dnsmasq".
5. Update resolv.conf ("netconfig update") to set the cache as the local DNS resolver.

### CentOS by Rogue Wave Software (formerly OpenLogic; uses NetworkManager)

1. Install the dnsmasq package ("sudo yum install dnsmasq").

2. Enable the dnsmasq service ("systemctl enable dnsmasq.service").
3. Start the dnsmasq service ("systemctl start dnsmasq.service").
4. Add "prepend domain-name-servers 127.0.0.1;" to "/etc/dhclient-eth0.conf".
5. Restart the network service ("service network restart") to set the cache as the local DNS resolver

**NOTE**

: The 'dnsmasq' package is only one of the many DNS caches that are available for Linux. Before you use it, check its suitability for your needs and that no other cache is installed.

## Client-side retries

DNS is primarily a UDP protocol. Because the UDP protocol doesn't guarantee message delivery, the DNS protocol itself handles retry logic. Each DNS client (operating system) can exhibit different retry logic depending on the creator's preference:

- Windows operating systems retry after one second and then again after another two, four, and another four seconds.
- The default Linux setup retries after five seconds. You should change this to retry five times at one-second intervals.

To check the current settings on a Linux virtual machine, 'cat /etc/resolv.conf', and look at the 'options' line, for example:

```
options timeout:1 attempts:5
```

The resolv.conf file is auto-generated and should not be edited. The specific steps that add the 'options' line vary by distribution:

### Ubuntu (uses resolvconf)

1. Add the options line to '/etc/resolvconf/resolv.conf.d/head'.
2. Run 'resolvconf -u' to update.

### SUSE (uses netconfig)

1. Add 'timeout:1 attempts:5' to the NETCONFIG\_DNS\_RESOLVER\_OPTIONS="" parameter in '/etc/sysconfig/network/config'.
2. Run 'netconfig update' to update.

### CentOS by Rogue Wave Software (formerly OpenLogic) (uses NetworkManager)

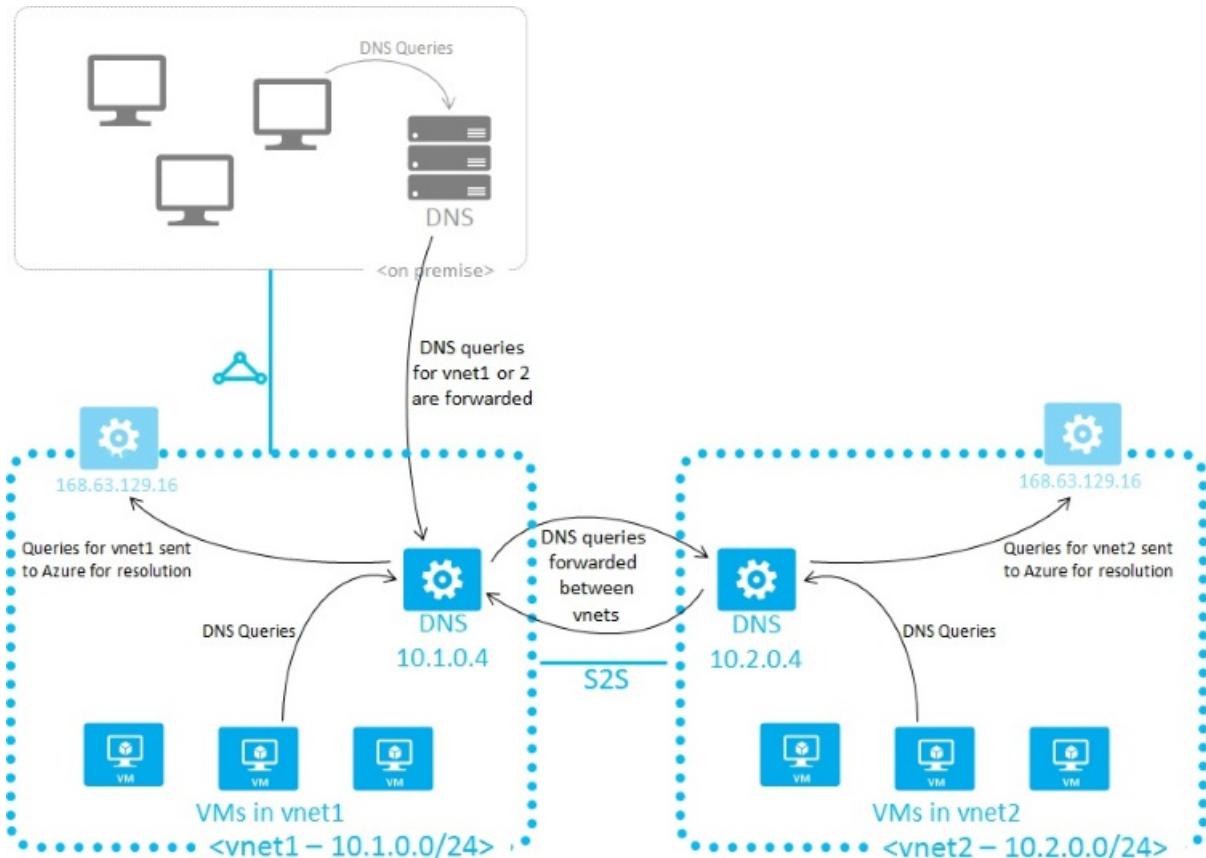
1. Add 'RES\_OPTIONS="timeout:1 attempts:5"' to '/etc/sysconfig/network'.
2. Run 'service network restart' to update.

## Name resolution using your own DNS server

Your name resolution needs may go beyond the features that Azure provides. For example, you might require DNS resolution between virtual networks. To cover this scenario, you can use your own DNS servers.

DNS servers within a virtual network can forward DNS queries to recursive resolvers of Azure to resolve hostnames that are in the same virtual network. For example, a DNS server that runs in Azure can respond to DNS queries for its own DNS zone files and forward all other queries to Azure. This functionality enables virtual machines to see both your entries in your zone files and hostnames that Azure provides (via the forwarder). Access to the recursive resolvers of Azure is provided via the virtual IP 168.63.129.16.

DNS forwarding also enables DNS resolution between virtual networks and enables your on-premises machines to resolve hostnames that Azure provides. To resolve a virtual machine's hostname, the DNS server virtual machine must reside in the same virtual network and be configured to forward hostname queries to Azure. Because the DNS suffix is different in each virtual network, you can use conditional forwarding rules to send DNS queries to the correct virtual network for resolution. The following image shows two virtual networks and an on-premises network doing DNS resolution between virtual networks by using this method:



When you use name resolution that Azure provides, the internal DNS suffix is provided to each virtual machine by using DHCP. When you use your own name resolution solution, this suffix is not supplied to virtual machines because the suffix interferes with other DNS architectures. To refer to machines by FQDN or to configure the suffix on your virtual machines, you can use PowerShell or the API to determine the suffix:

- For virtual networks that are managed by Azure Resource Manager, the suffix is available via the [network interface card](#) resource. You can also run the `azurerm network public-ip show <resource group> <ip name>` command to display the details of your public IP, which includes the FQDN of the NIC.

If forwarding queries to Azure doesn't suit your needs, you need to provide your own DNS solution. Your DNS solution needs to:

- Provide appropriate hostname resolution, for example via [DDNS](#). If you use DDNS, you might need to disable DNS record scavenging. DHCP leases of Azure are very long and scavenging may remove DNS records prematurely.
- Provide appropriate recursive resolution to allow resolution of external domain names.
- Be accessible (TCP and UDP on port 53) from the clients it serves and be able to access the Internet.
- Be secured against access from the Internet to mitigate threats posed by external agents.

#### NOTE

For best performance, when you use virtual machines in Azure DNS servers, disable IPv6 and assign an [Instance-Level Public IP](#) to each DNS server virtual machine.

# Create virtual network interface cards and use internal DNS for VM name resolution on Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article shows you how to set static internal DNS names for Linux VMs using virtual network interface cards (vNics) and DNS label names with the Azure CLI. Static DNS names are used for permanent infrastructure services like a Jenkins build server, which is used for this document, or a Git server.

The requirements are:

- [an Azure account](#)
- [SSH public and private key files](#)

## Quick commands

If you need to quickly accomplish the task, the following section details the commands needed. More detailed information and context for each step can be found in the rest of the document, [starting here](#). To perform these steps, you need the latest [Azure CLI](#) installed and logged in to an Azure account using [az login](#).

Pre-Requirements: Resource Group, virtual network and subnet, Network Security Group with SSH inbound.

### Create a virtual network interface card with a static internal DNS name

Create the vNic with [az network nic create](#). The `--internal-dns-name` CLI flag is for setting the DNS label, which provides the static DNS name for the virtual network interface card (vNic). The following example creates a vNic named `myNic`, connects it to the `myVnet` virtual network, and creates an internal DNS name record called `jenkins`:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic \
--vnet-name myVnet \
--subnet mySubnet \
--internal-dns-name jenkins
```

### Deploy a VM and connect the vNic

Create a VM with [az vm create](#). The `--nics` flag connects the vNic to the VM during the deployment to Azure. The following example creates a VM named `myVM` with Azure Managed Disks and attaches the vNic named `myNic` from the preceding step:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--nics myNic \
--image UbuntuLTS \
--admin-username azureuser \
--ssh-key-value ~/.ssh/id_rsa.pub
```

## Detailed walkthrough

A full continuous integration and continuous deployment (CiCd) infrastructure on Azure requires certain servers to be static or long-lived servers. It is recommended that Azure assets like the virtual networks and Network Security Groups are static and long lived resources that are rarely deployed. Once a virtual network has been deployed, it can be reused by new deployments without any adverse affects to the infrastructure. You can later add a Git repository server or a Jenkins automation server delivers CiCd to this virtual network for your development or test environments.

Internal DNS names are only resolvable inside an Azure virtual network. Because the DNS names are internal, they are not resolvable to the outside internet, providing additional security to the infrastructure.

In the following examples, replace example parameter names with your own values. Example parameter names include `myResourceGroup`, `myNic`, and `myVM`.

## Create the resource group

First, create the resource group with [az group create](#). The following example creates a resource group named `myResourceGroup` in the `westus` location:

```
az group create --name myResourceGroup --location westus
```

## Create the virtual network

The next step is to build a virtual network to launch the VMs into. The virtual network contains one subnet for this walkthrough. For more information on Azure virtual networks, see [Create a virtual network](#).

Create the virtual network with [az network vnet create](#). The following example creates a virtual network named `myVnet` and subnet named `mySubnet`:

```
az network vnet create \
--resource-group myResourceGroup \
--name myVnet \
--address-prefix 192.168.0.0/16 \
--subnet-name mySubnet \
--subnet-prefix 192.168.1.0/24
```

## Create the Network Security Group

Azure Network Security Groups are equivalent to a firewall at the network layer. For more information about Network Security Groups, see [How to create NSGs in the Azure CLI](#).

Create the network security group with [az network nsg create](#). The following example creates a network security group named `myNetworkSecurityGroup`:

```
az network nsg create \
--resource-group myResourceGroup \
--name myNetworkSecurityGroup
```

## Add an inbound rule to allow SSH

Add an inbound rule for the network security group with [az network nsg rule create](#). The following example creates a rule named `myRuleAllowSSH`:

```
az network nsg rule create \
--resource-group myResourceGroup \
--nsg-name myNetworkSecurityGroup \
--name myRuleAllowSSH \
--protocol tcp \
--direction inbound \
--priority 1000 \
--source-address-prefix '*' \
--source-port-range '*' \
--destination-address-prefix '*' \
--destination-port-range 22 \
--access allow
```

## Associate the subnet with the Network Security Group

To associate the subnet with the Network Security Group, use [az network vnet subnet update](#). The following example associates the subnet name `mySubnet` with the Network Security Group named `myNetworkSecurityGroup`:

```
az network vnet subnet update \
--resource-group myResourceGroup \
--vnet-name myVnet \
--name mySubnet \
--network-security-group myNetworkSecurityGroup
```

## Create the virtual network interface card and static DNS names

Azure is very flexible, but to use DNS names for VM name resolution, you need to create virtual network interface cards (vNics) that include a DNS label. vNics are important as you can reuse them by connecting them to different VMs over the infrastructure lifecycle. This approach keeps the vNic as a static resource while the VMs can be temporary. By using DNS labeling on the vNic, we are able to enable simple name resolution from other VMs in the VNet. Using resolvable names enables other VMs to access the automation server by the DNS name `Jenkins` or the Git server as `gitrepo`.

Create the vNic with [az network nic create](#). The following example creates a vNic named `myNic`, connects it to the `myVnet` virtual network named `myVnet`, and creates an internal DNS name record called `jenkins`:

```
az network nic create \
--resource-group myResourceGroup \
--name myNic \
--vnet-name myVnet \
--subnet mySubnet \
--internal-dns-name jenkins
```

## Deploy the VM into the virtual network infrastructure

We now have a virtual network and subnet, a Network Security Group acting as a firewall to protect our subnet by blocking all inbound traffic except port 22 for SSH, and a vNic. You can now deploy a VM inside this existing network infrastructure.

Create a VM with [az vm create](#). The following example creates a VM named `myVM` with Azure Managed Disks and attaches the vNic named `myNic` from the preceding step:

```
az vm create \
--resource-group myResourceGroup \
--name myVM \
--nics myNic \
--image UbuntuLTS \
--admin-username azureuser \
--ssh-key-value ~/.ssh/id_rsa.pub
```

By using the CLI flags to call out existing resources, we instruct Azure to deploy the VM inside the existing network. To reiterate, once a VNet and subnet have been deployed, they can be left as static or permanent resources inside your Azure region.

## Next steps

- [Create your own custom environment for a Linux VM using Azure CLI commands directly](#)
- [Create a Linux VM on Azure using templates](#)

# Azure Virtual Machines security overview

9/21/2022 • 7 minutes to read • [Edit Online](#)

This article provides an overview of the core Azure security features that can be used with virtual machines.

You can use Azure Virtual Machines to deploy a wide range of computing solutions in an agile way. The service supports Microsoft Windows, Linux, Microsoft SQL Server, Oracle, IBM, SAP, and Azure BizTalk Services. So you can deploy any workload and any language on nearly any operating system.

An Azure virtual machine gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs the virtual machine. You can build and deploy your applications with the assurance that your data is protected and safe in highly secure datacenters.

With Azure, you can build security-enhanced, compliant solutions that:

- Protect your virtual machines from viruses and malware.
- Encrypt your sensitive data.
- Secure network traffic.
- Identify and detect threats.
- Meet compliance requirements.

## Antimalware

With Azure, you can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky. This software helps protect your virtual machines from malicious files, adware, and other threats.

Microsoft Antimalware for Azure Cloud Services and Virtual Machines is a real-time protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware for Azure provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems.

Microsoft Antimalware for Azure is a single-agent solution for applications and tenant environments. It's designed to run in the background without human intervention. You can deploy protection based on the needs of your application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring.

Learn more about [Microsoft Antimalware for Azure](#) and the core features available.

Learn more about antimalware software to help protect your virtual machines:

- [Deploying Antimalware Solutions on Azure Virtual Machines](#)
- [How to install and configure Trend Micro Deep Security as a service on a Windows VM](#)
- [How to install and configure Symantec Endpoint Protection on a Windows VM](#)
- [Security solutions in the Azure Marketplace](#)

For even more powerful protection, consider using [Windows Defender Advanced Threat Protection](#). With Windows Defender ATP, you get:

- [Attack surface reduction](#)
- [Next generation protection](#)
- [Endpoint protection and response](#)
- [Automated investigation and remediation](#)

- [Secure score](#)
- [Advanced hunting](#)
- [Management and APIs](#)
- [Microsoft Threat Protection](#)

Learn more:

- [Get Started with WDATP](#)
- [Overview of WDATP capabilities](#)

## Hardware security module

Improving key security can enhance encryption and authentication protections. You can simplify the management and security of your critical secrets and keys by storing them in Azure Key Vault.

Key Vault provides the option to store your keys in hardware security modules (HSMs) certified to FIPS 140-2 Level 2 standards. Your SQL Server encryption keys for backup or [transparent data encryption](#) can all be stored in Key Vault with any keys or secrets from your applications. Permissions and access to these protected items are managed through [Azure Active Directory](#).

Learn more:

- [What is Azure Key Vault?](#)
- [Azure Key Vault blog](#)

## Virtual machine disk encryption

Azure Disk Encryption is a new capability for encrypting your Windows and Linux virtual machine disks. Azure Disk Encryption uses the industry-standard [BitLocker](#) feature of Windows and the [dm-crypt](#) feature of Linux to provide volume encryption for the OS and the data disks.

The solution is integrated with Azure Key Vault to help you control and manage the disk encryption keys and secrets in your key vault subscription. It ensures that all data in the virtual machine disks are encrypted at rest in Azure Storage.

Learn more:

- [Azure Disk Encryption for Linux VMs](#) and [Azure Disk Encryption for Windows VMs](#)
- [Quickstart: Encrypt a Windows IaaS VM with Azure PowerShell](#)

## Virtual machine backup

Azure Backup is a scalable solution that helps protect your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. With Azure Backup, your virtual machines running Windows and Linux are protected.

Learn more:

- [What is Azure Backup?](#)
- [Azure Backup service FAQ](#)

## Azure Site Recovery

An important part of your organization's BCDR strategy is figuring out how to keep corporate workloads and apps running when planned and unplanned outages occur. Azure Site Recovery helps orchestrate replication, failover, and recovery of workloads and apps so that they're available from a secondary location if your primary

location goes down.

Site Recovery:

- **Simplifies your BCDR strategy:** Site Recovery makes it easy to handle replication, failover, and recovery of multiple business workloads and apps from a single location. Site Recovery orchestrates replication and failover but doesn't intercept your application data or have any information about it.
- **Provides flexible replication:** By using Site Recovery, you can replicate workloads running on Hyper-V virtual machines, VMware virtual machines, and Windows/Linux physical servers.
- **Supports failover and recovery:** Site Recovery provides test failovers to support disaster recovery drills without affecting production environments. You can also run planned failovers with a zero-data loss for expected outages, or unplanned failovers with minimal data loss (depending on replication frequency) for unexpected disasters. After failover, you can fail back to your primary sites. Site Recovery provides recovery plans that can include scripts and Azure Automation workbooks so that you can customize failover and recovery of multi-tier applications.
- **Eliminates secondary datacenters:** You can replicate to a secondary on-premises site, or to Azure. Using Azure as a destination for disaster recovery eliminates the cost and complexity of maintaining a secondary site. Replicated data is stored in Azure Storage.
- **Integrates with existing BCDR technologies:** Site Recovery partners with other applications' BCDR features. For example, you can use Site Recovery to help protect the SQL Server back end of corporate workloads. This includes native support for SQL Server Always On to manage the failover of availability groups.

Learn more:

- [What is Azure Site Recovery?](#)
- [How does Azure Site Recovery work?](#)
- [What workloads are protected by Azure Site Recovery?](#)

## Virtual networking

Virtual machines need network connectivity. To support that requirement, Azure requires virtual machines to be connected to an Azure virtual network.

An Azure virtual network is a logical construct built on top of the physical Azure network fabric. Each logical Azure virtual network is isolated from all other Azure virtual networks. This isolation helps ensure that network traffic in your deployments is not accessible to other Microsoft Azure customers.

Learn more:

- [Azure network security overview](#)
- [Virtual Network overview](#)
- [Networking features and partnerships for enterprise scenarios](#)

## Security policy management and reporting

Microsoft Defender for Cloud helps you prevent, detect, and respond to threats. Defender for Cloud gives you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions. It helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Defender for Cloud helps you optimize and monitor the security of your virtual machines by:

- Providing [security recommendations](#) for the virtual machines. Example recommendations include: apply system updates, configure ACLs endpoints, enable antimalware, enable network security groups, and apply

disk encryption.

- Monitoring the state of your virtual machines.

Learn more:

- [Introduction to Microsoft Defender for Cloud](#)
- [Microsoft Defender for Cloud frequently asked questions](#)
- [Microsoft Defender for Cloud planning and operations](#)

## Compliance

Azure Virtual Machines is certified for FISMA, FedRAMP, HIPAA, PCI DSS Level 1, and other key compliance programs. This certification makes it easier for your own Azure applications to meet compliance requirements and for your business to address a wide range of domestic and international regulatory requirements.

Learn more:

- [Microsoft Trust Center: Compliance](#)
- [Trusted Cloud: Microsoft Azure Security, Privacy, and Compliance](#)

## Confidential Computing

While confidential computing is not technically part of virtual machine security, the topic of virtual machine security belongs to the higher-level subject of "compute" security. Confidential computing belongs within the category of "compute" security.

Confidential computing ensures that when data is "in the clear," which is required for efficient processing, the data is protected inside a Trusted Execution Environment

[https://en.wikipedia.org/wiki/Trusted\\_execution\\_environment](https://en.wikipedia.org/wiki/Trusted_execution_environment) (TEE - also known as an enclave), an example of which is shown in the figure below.

TEEs ensure there is no way to view data or the operations inside from the outside, even with a debugger. They even ensure that only authorized code is permitted to access data. If the code is altered or tampered, the operations are denied and the environment disabled. The TEE enforces these protections throughout the execution of code within it.

Learn more:

- [Introducing Azure confidential computing](#)
- [Azure confidential computing](#)

## Next steps

Learn about [security best practices](#) for VMs and operating systems.

# Security recommendations for virtual machines in Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article contains security recommendations for Azure Virtual Machines. Follow these recommendations to help fulfill the security obligations described in our model for shared responsibility. The recommendations will also help you improve overall security for your web app solutions. For more information about what Microsoft does to fulfill service-provider responsibilities, see [Shared responsibilities for cloud computing](#).

Some of this article's recommendations can be automatically addressed by Microsoft Defender for Cloud. Microsoft Defender for Cloud is the first line of defense for your resources in Azure. It periodically analyzes the security state of your Azure resources to identify potential security vulnerabilities. It then recommends how to address the vulnerabilities. For more information, see [Security recommendations in Microsoft Defender for Cloud](#).

For general information about Microsoft Defender for Cloud, see [What is Microsoft Defender for Cloud?](#).

## General

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
When you build custom VM images, apply the latest updates.	Before you create images, install the latest updates for the operating system and for all applications that will be part of your image.	-
Keep your VMs current.	You can use the <a href="#">Update Management</a> solution in Azure Automation to manage operating system updates for your Windows and Linux computers in Azure.	Yes
Back up your VMs.	<a href="#">Azure Backup</a> helps protect your application data and has minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. Azure Backup protects your VMs that run Windows and Linux.	-
Use multiple VMs for greater resilience and availability.	If your VM runs applications that must be highly available, use multiple VMs or <a href="#">availability sets</a> .	-

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Adopt a business continuity and disaster recovery (BCDR) strategy.	Azure Site Recovery allows you to choose from different options designed to support business continuity. It supports different replication and failover scenarios. For more information, see <a href="#">About Site Recovery</a> .	-

## Data security

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Encrypt operating system disks.	<a href="#">Azure Disk Encryption</a> helps you encrypt your Windows and Linux IaaS VM disks. Without the necessary keys, the contents of encrypted disks are unreadable. Disk encryption protects stored data from unauthorized access that would otherwise be possible if the disk were copied.	Yes
Encrypt data disks.	<a href="#">Azure Disk Encryption</a> helps you encrypt your Windows and Linux IaaS VM disks. Without the necessary keys, the contents of encrypted disks are unreadable. Disk encryption protects stored data from unauthorized access that would otherwise be possible if the disk were copied.	-
Limit installed software.	Limit installed software to what is required to successfully apply your solution. This guideline helps reduce your solution's attack surface.	-
Use antivirus or antimalware.	In Azure, you can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky. This software helps protect your VMs from malicious files, adware, and other threats. You can deploy Microsoft Antimalware based on your application workloads. Microsoft Antimalware is available for Windows machines only. Use either basic secure-by-default or advanced custom configuration. For more information, see <a href="#">Microsoft Antimalware for Azure Cloud Services and Virtual Machines</a> .	-

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Securely store keys and secrets.	Simplify the management of your secrets and keys by providing your application owners with a secure, centrally managed option. This management reduces the risk of an accidental compromise or leak. Azure Key Vault can securely store your keys in hardware security modules (HSMs) that are certified to FIPS 140-2 Level 2. If you need to use FIPs 140.2 Level 3 to store your keys and secrets, you can use <a href="#">Azure Dedicated HSM</a> .	-

## Identity and access management

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Centralize VM authentication.	You can centralize the authentication of your Windows and Linux VMs by using <a href="#">Azure Active Directory authentication</a> .	-

## Monitoring

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Monitor your VMs.	You can use <a href="#">Azure Monitor for VMs</a> to monitor the state of your Azure VMs and virtual machine scale sets. Performance issues with a VM can lead to service disruption, which violates the security principle of availability.	-

## Networking

RECOMMENDATION	COMMENTS	DEFENDER FOR CLOUD
Restrict access to management ports.	Attackers scan public cloud IP ranges for open management ports and attempt "easy" attacks like common passwords and known unpatched vulnerabilities. You can use <a href="#">just-in-time (JIT) VM access</a> to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy connections to VMs when they're needed.	-
Limit network access.	Network security groups allow you to restrict network access and control the number of exposed endpoints. For more information, see <a href="#">Create, change, or delete a network security group</a> .	-

## Next steps

Check with your application provider to learn about additional security requirements. For more information about developing secure applications, see [Secure-development documentation](#).

# Trusted launch for Azure virtual machines

9/21/2022 • 11 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure offers trusted launch as a seamless way to improve the security of [generation 2](#) VMs. Trusted launch protects against advanced and persistent attack techniques. Trusted launch is composed of several, coordinated infrastructure technologies that can be enabled independently. Each technology provides another layer of defense against sophisticated threats.

## IMPORTANT

Trusted launch requires the creation of new virtual machines. You can't enable trusted launch on existing virtual machines that were initially created without it.

## Benefits

- Securely deploy virtual machines with verified boot loaders, OS kernels, and drivers.
- Securely protect keys, certificates, and secrets in the virtual machines.
- Gain insights and confidence of the entire boot chain's integrity.
- Ensure workloads are trusted and verifiable.

## Limitations

### VM size support:

- B-series
- DCsv2-series
- DCsv3-series, DCdsv3-series
- Dv4-series, Dsv4-series, Dsv3-series, Dsv2-series
- Dav4-series, Dasv4-series
- Ddv4-series, Ddsv4-series
- Dv5-series, Dsv5-series
- Ddv5-series, Ddsv5-series
- Dasv5-series, Dadsv5-series
- Ev5-series, Esv5-series
- Edv5-series, Edsv5-series
- Easv5-series, Eadsv5-series
- Ebsv5-series, Ebdsv5-series
- Eav4-series, Easv4-series
- Ev4-series, Esv4-series, Esv3-series
- Edv4-series, Edsv4-series
- Fsv2-series
- Lsv2-series

### OS support:

- Redhat Enterprise Linux 8.3, 8.4, 8.5 LVM

- SUSE Enterprise Linux 15 SP3
- Ubuntu Server 22.04 LTS
- Ubuntu Server 20.04 LTS
- Ubuntu Server 18.04 LTS
- Debian 11
- CentOS 8.3, 8.4
- Oracle Linux 8.3 LVM
- CBL-Mariner
- Windows Server 2022
- Windows Server 2019
- Windows Server 2016
- Windows 11 Pro
- Windows 11 Enterprise
- Windows 11 Enterprise multi-session
- Windows 10 Pro
- Windows 10 Enterprise
- Windows 10 Enterprise multi-session

**Regions:**

- All public regions

**Pricing:** No additional cost to existing VM pricing.

**The following features are not supported:**

- Azure Site Recovery
- Shared disk
- Ultra disk
- Managed image
- Nested Virtualization

## Secure boot

At the root of trusted launch is Secure Boot for your VM. This mode, which is implemented in platform firmware, protects against the installation of malware-based rootkits and boot kits. Secure Boot works to ensure that only signed operating systems and drivers can boot. It establishes a "root of trust" for the software stack on your VM. With Secure Boot enabled, all OS boot components (boot loader, kernel, kernel drivers) must be signed by trusted publishers. Both Windows and select Linux distributions support Secure Boot. If Secure Boot fails to authenticate that the image was signed by a trusted publisher, the VM will not be allowed to boot. For more information, see [Secure Boot](#).

## vTPM

Trusted launch also introduces vTPM for Azure VMs. This is a virtualized version of a hardware [Trusted Platform Module](#), compliant with the TPM2.0 spec. It serves as a dedicated secure vault for keys and measurements.

Trusted launch provides your VM with its own dedicated TPM instance, running in a secure environment outside the reach of any VM. The vTPM enables [attestation](#) by measuring the entire boot chain of your VM (UEFI, OS, system, and drivers).

Trusted launch uses the vTPM to perform remote attestation by the cloud. This is used for platform health checks and for making trust-based decisions. As a health check, trusted launch can cryptographically certify that your

VM booted correctly. If the process fails, possibly because your VM is running an unauthorized component, Microsoft Defender for Cloud will issue integrity alerts. The alerts include details on which components failed to pass integrity checks.

## Virtualization-based security

[Virtualization-based Security \(VBS\)](#) uses the hypervisor to create a secure and isolated region of memory. Windows uses these regions to run various security solutions with increased protection against vulnerabilities and malicious exploits. Trusted launch lets you enable Hypervisor Code Integrity (HVCI) and Windows Defender Credential Guard.

HVCI is a powerful system mitigation that protects Windows kernel-mode processes against injection and execution of malicious or unverified code. It checks kernel mode drivers and binaries before they run, preventing unsigned files from loading into memory. This ensures such executable code can't be modified once it is allowed to load. For more information about VBS and HVCI, see [Virtualization Based Security \(VBS\) and Hypervisor Enforced Code Integrity \(HVCI\)](#).

With trusted launch and VBS you can enable Windows Defender Credential Guard. This feature isolates and protects secrets so that only privileged system software can access them. It helps prevent unauthorized access to secrets and credential theft attacks, like Pass-the-Hash (PtH) attacks. For more information, see [Credential Guard](#).

## Microsoft Defender for Cloud integration

Trusted launch is integrated with Azure Defender for Cloud to ensure your VMs are properly configured. Azure Defender for Cloud will continually assess compatible VMs and issue relevant recommendations.

- **Recommendation to enable Secure Boot** - This Recommendation only applies for VMs that support trusted launch. Azure Defender for Cloud will identify VMs that can enable Secure Boot, but have it disabled. It will issue a low severity recommendation to enable it.
- **Recommendation to enable vTPM** - If your VM has vTPM enabled, Azure Defender for Cloud can use it to perform Guest Attestation and identify advanced threat patterns. If Azure Defender for Cloud identifies VMs that support trusted launch and have vTPM disabled, it will issue a low severity recommendation to enable it.
- **Recommendation to install guest attestation extension** - If your VM has secure boot and vTPM enabled but it doesn't have the guest attestation extension installed, Azure Defender for Cloud will issue a low severity recommendation to install the guest attestation extension on it. This extension allows Azure Defender for Cloud to proactively attest and monitor the boot integrity of your VMs. Boot integrity is attested via remote attestation.
- **Attestation health assessment or Boot Integrity Monitoring** - If your VM has Secure Boot and vTPM enabled and attestation extension installed, Azure Defender for Cloud can remotely validate that your VM booted in a healthy way. This is known as boot integrity monitoring. Azure Defender for Cloud issues an assessment, indicating the status of remote attestation. Currently boot integrity monitoring is supported for both Windows and Linux single virtual machines and uniform scale sets.

If your VMs are properly set up with trusted launch, Microsoft Defender for Cloud can detect and alert you of VM health problems.

- **Alert for VM attestation failure:** Microsoft Defender for Cloud will periodically perform attestation on your VMs. This also happens after your VM boots. If the attestation fails, it will trigger a medium severity alert. VM attestation can fail for the following reasons:
  - The attested information, which includes a boot log, deviates from a trusted baseline. This can indicate that untrusted modules have been loaded, and the OS may be compromised.
  - The attestation quote could not be verified to originate from the vTPM of the attested VM. This can indicate that malware is present and may be intercepting traffic to the vTPM.

#### **NOTE**

This alert is available for VMs with vTPM enabled and the Attestation extension installed. Secure Boot must be enabled for attestation to pass. Attestation will fail if Secure Boot is disabled. If you must disable Secure Boot, you can suppress this alert to avoid false positives.

- **Alert for Untrusted Linux Kernel module:** For trusted launch with secure boot enabled, it's possible for a VM to boot even if a kernel driver fails validation and is prohibited from loading. If this happens, Microsoft Defender for Cloud will issue a low severity alert. While there is no immediate threat, because the untrusted driver has not been loaded, these events should be investigated. Consider the following:
  - Which kernel driver failed? Am I familiar with this driver and expect it to be loaded?
  - Is this the exact version of the driver I am expecting? Are the driver binaries intact? If this is a 3rd party driver, did the vendor pass the OS compliance tests to get it signed?

## FAQ

Frequently asked questions about trusted launch.

### **Why should I use trusted launch? What does trusted launch guard against?**

Trusted launch guards against boot kits, rootkits, and kernel-level malware. These sophisticated types of malware run in kernel mode and remain hidden from users. For example:

- Firmware rootkits: these kits overwrite the firmware of the virtual machine's BIOS, so the rootkit can start before the OS.
- Boot kits: these kits replace the OS's bootloader so that the virtual machine loads the boot kit before the OS.
- Kernel rootkits: these kits replace a portion of the OS kernel so the rootkit can start automatically when the OS loads.
- Driver rootkits: these kits pretend to be one of the trusted drivers that OS uses to communicate with the virtual machine's components.

### **What are the differences between secure boot and measured boot?**

In secure boot chain, each step in the boot process checks a cryptographic signature of the subsequent steps. For example, the BIOS will check a signature on the loader, and the loader will check signatures on all the kernel objects that it loads, and so on. If any of the objects are compromised, the signature won't match, and the VM will not boot. For more information, see [Secure Boot](#). Measured boot does not halt the boot process, it measures or computes the hash of the next objects in the chain and stores the hashes in the Platform Configuration Registers (PCRs) on the vTPM. Measured boot records are used for boot integrity monitoring.

### **What happens when an integrity fault is detected?**

Trusted launch for Azure virtual machines is monitored for advanced threats. If such threats are detected, an alert will be triggered. Alerts are only available if [Defender for Cloud's enhanced security features](#) are enabled.

Defender for Cloud periodically performs attestation. If the attestation fails, a medium severity alert will be triggered. Trusted launch attestation can fail for the following reasons:

Trusted launch for Azure virtual machines is monitored for advanced threats. If such threats are detected, an alert will be triggered. Alerts are only available in the [Standard Tier](#) of Azure Defender for Cloud. Azure Defender for Cloud periodically performs attestation. If the attestation fails, a medium severity alert will be triggered. Trusted launch attestation can fail for the following reasons:

- The attested information, which includes a log of the Trusted Computing Base (TCB), deviates from a trusted baseline (like when Secure Boot is enabled). This can indicate that untrusted modules have been loaded and the OS may be compromised.

- The attestation quote could not be verified to originate from the vTPM of the attested VM. This can indicate that malware is present and may be intercepting traffic to the TPM.
- The attestation extension on the VM is not responding. This can indicate a denial-of-service attack by malware, or an OS admin.

## How does trusted launch compare to Hyper-V Shielded VM?

Hyper-V Shielded VM is currently available on Hyper-V only. [Hyper-V Shielded VM](#) is typically deployed in conjunction with Guarded Fabric. A Guarded Fabric consists of a Host Guardian Service (HGS), one or more guarded hosts, and a set of Shielded VMs. Hyper-V Shielded VMs are intended for use in fabrics where the data and state of the virtual machine must be protected from both fabric administrators and untrusted software that might be running on the Hyper-V hosts. Trusted launch on the other hand can be deployed as a standalone virtual machine or virtual machine scale sets on Azure without additional deployment and management of HGS. All of the trusted launch features can be enabled with a simple change in deployment code or a checkbox on the Azure portal.

## Does trusted launch support Azure Compute Gallery?

Trusted launch now allows images to be created and shared through the Azure Compute Gallery (formerly Shared Image Gallery). The image source can be an existing Azure VM which is either generalized or specialized, an existing managed disk or a snapshot, a VHD or an image version from another gallery. To deploy a Trusted Launch VM from an Azure Compute Gallery image version see [trusted launch VM](#).

## Does trusted launch support Azure Backup?

Trusted launch now supports Azure Backup. For more information, see [Support matrix for Azure VM backup](#).

## Does trusted launch support ephemeral OS disks?

Trusted launch supports ephemeral OS disks. Note that, while using ephemeral disks for Trusted Launch VMs, keys and secrets generated or sealed by the vTPM after the creation of the VM may not be persisted across operations like reimaging and platform events like service healing. For more information, see [Trusted Launch for Ephemeral OS disks \(Preview\)](#).

## How can I find VM sizes that support Trusted launch?

See the list of [Generation 2 VM sizes supporting Trusted launch](#).

The following commands can be used to check if a [Generation 2 VM Size](#) does not support Trusted launch.

### CLI

```
subscription=<yourSubID>
region="westus"
vmSize="Standard_NC12s_v3"

az vm list-skus --resource-type virtualMachines --location $region --query "[?
name=='$vmSize'].capabilities" --subscription $subscription
```

### PowerShell

```
$region = "southeastasia"
$vmSize = "Standard_M64"
(Get-AzComputeResourceSku | where {$_.Locations.Contains($region) -and ($_.Name -eq $vmSize) })
[0].Capabilities
```

The response will be similar to the following form. `TrustedLaunchDisabled True` in the output indicates that the Generation 2 VM size does not support Trusted launch. If it's a Generation 2 VM size and `TrustedLaunchDisabled` is not part of the output, it implies that Trusted launch is supported for that VM size.

Name	Value
---	----
MaxResourceVolumeMB	8192000
OSVhdSizeMB	1047552
vCPUs	64
MemoryPreservingMaintenanceSupported	False
HyperVGenerations	V1,V2
MemoryGB	1000
MaxDataDiskCount	64
CpuArchitectureType	x64
MaxWriteAcceleratorDisksAllowed	8
LowPriorityCapable	True
PremiumIO	True
VMDeploymentTypes	IaaS
vCPUsAvailable	64
ACUs	160
vCPUsPerCore	2
CombinedTempDiskAndCachedIOPS	80000
CombinedTempDiskAndCachedReadBytesPerSecond	838860800
CombinedTempDiskAndCachedWriteBytesPerSecond	838860800
CachedDiskBytes	1318554959872
UncachedDiskIOPS	40000
UncachedDiskBytesPerSecond	1048576000
EphemeralOSDiskSupported	True
EncryptionAtHostSupported	True
CapacityReservationSupported	False
TrustedLaunchDisabled	True
AcceleratedNetworkingEnabled	True
RdmaEnabled	False
MaxNetworkInterfaces	8

## What is VM Guest State (VMGS)?

VM Guest State (VMGS) is specific to Trusted Launch VM. It is a blob that is managed by Azure and contains the unified extensible firmware interface (UEFI) secure boot signature databases and other security information. The lifecycle of the VMGS blob is tied to that of the OS Disk.

## Next steps

Deploy a [trusted launch VM](#).

# Deploy a VM with trusted launch enabled

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

**Trusted launch** is a way to improve the security of [generation 2](#) VMs. Trusted launch protects against advanced and persistent attack techniques by combining infrastructure technologies like vTPM and secure boot.

## Prerequisites

- You need to [onboard your subscription to Microsoft Defender for Cloud](#) if it isn't already. Microsoft Defender for Cloud has a free tier, which offers very useful insights for various Azure and Hybrid resources. Trusted launch leverages Defender for Cloud to surface multiple recommendations regarding VM health.
- Assign Azure policies initiatives to your subscription. These policy initiatives need to be assigned only once per subscription. This will automatically install all required extensions on all supported VMs.
  - Configure prerequisites to enable Guest Attestation on Trusted Launch enabled VMs
  - Configure machines to automatically install the Azure Monitor and Azure Security agents on virtual machines

## Deploy a trusted launch VM

Create a virtual machine with trusted launch enabled. Choose an option below:

- [Portal](#)
- [CLI](#)
- [PowerShell](#)
- [Template](#)

1. Sign in to the Azure [portal](#).
2. Search for **Virtual Machines**.
3. Under **Services**, select **Virtual machines**.
4. In the **Virtual machines** page, select **Add**, and then select **Virtual machine**.
5. Under **Project details**, make sure the correct subscription is selected.
6. Under **Resource group**, select **Create new** and type a name for your resource group or select an existing resource group from the dropdown.
7. Under **Instance details**, type a name for the virtual machine name and choose a region that supports [trusted launch](#).
8. For **Security type** select **Trusted launch virtual machines**. This will make two more options appear - **Secure boot** and **vTPM**. Select the appropriate options for your deployment.

Instance details

Virtual machine name *	myTVM
Region *	(US) East US
Availability options	Availability zone
Availability zone *	1
Security type	Trusted launch virtual machines
Secure boot	<input checked="" type="checkbox"/>
vTPM	<input checked="" type="checkbox"/>

- Under **Image**, select an image from the **Recommended Gen 2 images compatible with Trusted launch**. For a list, see [images that supports trusted launch](#).

**TIP**

If you don't see the Gen 2 version of the image you want in the drop-down, select **See all images** and then change the **Security type** filter to **Trusted Launch**.

- Select a VM size that supports trusted launch. See the list of [supported sizes](#).
- Fill in the **Administrator account** information and then **Inbound port rules**.
- At the bottom of the page, select **Review + Create**
- On the **Create a virtual machine** page, you can see the details about the VM you are about to deploy. Once validation shows as passed, select **Create**.

**Basics**

Subscription	myAzureSubscription
Resource group	myresourcegroup
Virtual machine name	myTrustedVM
Region	East US
Availability options	No infrastructure redundancy required
Security type	Trusted launch virtual machines
Secure boot	Yes
vTPM	Yes
Image	Windows Server 2019 Datacenter - Gen2
Size	Standard B2s (2 vcpus, 4 GiB memory)
Username	azureuser
Public inbound ports	RDP
Already have a Windows license?	No

It will take a few minutes for your VM to be deployed.

## Deploy a trusted launch VM from an Azure Compute Gallery image

- [Portal](#)
- [CLI](#)
- [PowerShell](#)

- Sign in to the Azure [portal](#).
- To create an Azure Compute Gallery Image from a VM, open an existing Trusted launch VM and select **Capture**.

3. In the Create an Image page that follows, allow the image to be shared to the gallery as a VM image version. Creation of Managed Images is not supported for Trusted Launch VMs.
4. Create a new target Azure Compute Gallery or select an existing gallery.
5. Select the **Operating system state** as either **Generalized** or **Specialized**. If you want to create a generalized image, ensure that you [generalize the VM to remove machine specific information](#) before selecting this option. If Bitlocker based encryption is enabled on your Trusted launch Windows VM, you may not be able to generalize the same.
6. Create a new image definition by providing a name, publisher, offer and SKU details. The **Security Type** of the image definition should already be set to **Trusted launch**.
7. Provide a version number for the image version.
8. Modify replication options if required.
9. At the bottom of the **Create an Image** page, select **Review + Create** and when validation shows as passed, select **Create**.
10. Once the image version is created, go the image version directly. Alternatively, you can navigate to the required image version through the image definition.
11. On the **VM image version** page, select the **+ Create VM** to land on the Create a virtual machine page.
12. In the Create a virtual machine page, under **Resource group**, select **Create new** and type a name for your resource group or select an existing resource group from the dropdown.
13. Under **Instance details**, type a name for the virtual machine name and choose a region that supports [trusted launch](#).
14. The image and the security type are already populated based on the selected image version. The **Secure Boot** and **vTPM** checkboxes are enabled by default.
15. Fill in the **Administrator account** information and then **Inbound port rules**.
16. At the bottom of the page, select **Review + Create**
17. On the **Create a virtual machine** page, you can see the details about the VM you are about to deploy. Once validation shows as passed, select **Create**.

In case you want to use either a managed disk or a managed disk snapshot as a source of the image version (instead of a trusted launch VM), then use the following steps

1. Sign in to the [portal](#)
2. Search for **VM Image Versions** and select **Create**
3. Provide the subscription, resource group, region and image version number
4. Select the source as **Disks and/or Snapshots**
5. Select the OS disk as a managed disk or a managed disk snapshot from the dropdown list
6. Select a **Target Azure Compute Gallery** to create and share the image. If no gallery exists, create a new gallery.
7. Select the **Operating system state** as either **Generalized** or **Specialized**. If you want to create a generalized image, ensure that you generalize the disk or snapshot to remove machine specific information.
8. For the **Target VM Image Definition** select **Create new**. In the window that opens, select an image definition name and ensure that the **Security type** is set to **Trusted launch**. Provide the publisher, offer and SKU information and select **OK**.
9. The **Replication** tab can be used to set the replica count and target regions for image replication, if required.
10. The **Encryption** tab can also be used to provide SSE encryption related information, if required.
11. Select **Create** in the **Review + create** tab to create the image
12. Once the image version is successfully created, select the **+ Create VM** to land on the Create a virtual machine page.
13. Please follow steps 12 to 17 as mentioned earlier to create a trusted launch VM using this image version

## Verify or update your settings

For VMs created with trusted launch enabled, you can view the trusted launch configuration by visiting the [Overview](#) page for the VM in the portal. The **Properties** tab will show the status of Trusted Launch features:

### Security type

Security type	Trusted launch
Secure boot	Enabled
vTPM	Enabled

To change the trusted launch configuration, in the left menu, select **Configuration** under the **Settings** section. You can enable or disable Secure Boot and vTPM from the **Trusted Launch Security types** section. Select **Save** at the top of the page when you are done.

#### Security type

The different levels of security available for your virtual machines. Standard offers basic protection at no extra costs. Trusted launch virtual machines provide additional security features on Gen 2 virtual machines to protect against persistent and advanced attacks. [Learn more about trusted launch](#) ↗

- Secure boot ⓘ
- vTPM ⓘ

If the VM is running, you will receive a message that the VM will be restarted. Select **Yes** then wait for the VM to restart for changes to take effect.

## Next steps

Learn more about [trusted launch](#) and [Generation 2 VMs](#).

# Secure and use policies on virtual machines in Azure

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

It's important to keep your virtual machine (VM) secure for the applications that you run. Securing your VMs can include one or more Azure services and features that cover secure access to your VMs and secure storage of your data. This article provides information that enables you to keep your VM and applications secure.

## Antimalware

The modern threat landscape for cloud environments is dynamic, increasing the pressure to maintain effective protection in order to meet compliance and security requirements. [Microsoft Antimalware for Azure](#) is a free real-time protection capability that helps identify and remove viruses, spyware, and other malicious software. Alerts can be configured to notify you when known malicious or unwanted software attempts to install itself or run on your VM. It is not supported on VMs running Linux or Windows Server 2008.

## Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats to your VMs. Defender for Cloud provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Defender for Cloud's just-in-time access can be applied across your VM deployment to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy access to connect to VMs when needed. When just-in-time is enabled and a user requests access to a VM, Defender for Cloud checks what permissions the user has for the VM. If they have the correct permissions, the request is approved and Defender for Cloud automatically configures the Network Security Groups (NSGs) to allow inbound traffic to the selected ports for a limited amount of time. After the time has expired, Defender for Cloud restores the NSGs to their previous states.

## Encryption

Two encryption methods are offered for managed disks. Encryption at the OS-level, which is Azure Disk Encryption, and encryption at the platform-level, which is server-side encryption.

### Server-side encryption

Azure managed disks automatically encrypt your data by default when persisting it to the cloud. Server-side encryption protects your data and helps you meet your organizational security and compliance commitments. Data in Azure managed disks is encrypted transparently using 256-bit [AES encryption](#), one of the strongest block ciphers available, and is FIPS 140-2 compliant.

Encryption does not impact the performance of managed disks. There is no additional cost for the encryption.

You can rely on platform-managed keys for the encryption of your managed disk, or you can manage encryption using your own keys. If you choose to manage encryption with your own keys, you can specify a *customer-managed key* to use for encrypting and decrypting all data in managed disks.

To learn more about server-side encryption, refer to either the articles for [Windows](#) or [Linux](#).

### Azure Disk Encryption

For enhanced [Windows VM](#) and [Linux VM](#) security and compliance, virtual disks in Azure can be encrypted.

Virtual disks on Windows VMs are encrypted at rest using BitLocker. Virtual disks on Linux VMs are encrypted at rest using dm-crypt.

There is no charge for encrypting virtual disks in Azure. Cryptographic keys are stored in Azure Key Vault using software-protection, or you can import or generate your keys in Hardware Security Modules (HSMs) certified to FIPS 140-2 level 2 standards. These cryptographic keys are used to encrypt and decrypt virtual disks attached to your VM. You retain control of these cryptographic keys and can audit their use. An Azure Active Directory service principal provides a secure mechanism for issuing these cryptographic keys as VMs are powered on and off.

## Key Vault and SSH Keys

Secrets and certificates can be modeled as resources and provided by [Key Vault](#). You can use Azure PowerShell to create key vaults for [Windows VMs](#) and the Azure CLI for [Linux VMs](#). You can also create keys for encryption.

Key vault access policies grant permissions to keys, secrets, and certificates separately. For example, you can give a user access to only keys, but no permissions for secrets. However, permissions to access keys or secrets or certificates are at the vault level. In other words, [key vault access policy](#) does not support object level permissions.

When you connect to VMs, you should use public-key cryptography to provide a more secure way to sign in to them. This process involves a public and private key exchange using the secure shell (SSH) command to authenticate yourself rather than a username and password. Passwords are vulnerable to brute-force attacks, especially on Internet-facing VMs such as web servers. With a secure shell (SSH) key pair, you can create a [Linux VM](#) that uses SSH keys for authentication, eliminating the need for passwords to sign-in. You can also use SSH keys to connect from a [Windows VM](#) to a Linux VM.

## Managed identities for Azure resources

A common challenge when building cloud applications is how to manage the credentials in your code for authenticating to cloud services. Keeping the credentials secure is an important task. Ideally, the credentials never appear on developer workstations and aren't checked into source control. Azure Key Vault provides a way to securely store credentials, secrets, and other keys, but your code has to authenticate to Key Vault to retrieve them.

The managed identities for Azure resources feature in Azure Active Directory (Azure AD) solves this problem. The feature provides Azure services with an automatically managed identity in Azure AD. You can use the identity to authenticate to any service that supports Azure AD authentication, including Key Vault, without any credentials in your code. Your code that's running on a VM can request a token from two endpoints that are accessible only from within the VM. For more detailed information about this service, review the [managed identities for Azure resources](#) overview page.

## Policies

[Azure policies](#) can be used to define the desired behavior for your organization's [Windows VMs](#) and [Linux VMs](#). By using policies, an organization can enforce various conventions and rules throughout the enterprise. Enforcement of the desired behavior can help mitigate risk while contributing to the success of the organization.

## Azure role-based access control

Using [Azure role-based access control \(Azure RBAC\)](#), you can segregate duties within your team and grant only the amount of access to users on your VM that they need to perform their jobs. Instead of giving everybody unrestricted permissions on the VM, you can allow only certain actions. You can configure access control for the VM in the [Azure portal](#), using the [Azure CLI](#), or [Azure PowerShell](#).

## Next steps

- Walk through the steps to monitor virtual machine security by using Microsoft Defender for Cloud for [Linux](#) or [Windows](#).

# Azure Policy Regulatory Compliance controls for Azure Virtual Machines

9/21/2022 • 158 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

[Regulatory Compliance in Azure Policy](#) provides Microsoft created and managed initiative definitions, known as *built-ins*, for the **compliance domains** and **security controls** related to different compliance standards. This page lists the **compliance domains** and **security controls** for Azure Virtual Machines. You can assign the built-ins for a **security control** individually to help make your Azure resources compliant with the specific standard.

The title of each built-in policy definition links to the policy definition in the Azure portal. Use the link in the **Policy Version** column to view the source on the [Azure Policy GitHub repo](#).

## IMPORTANT

Each control is associated with one or more [Azure Policy](#) definitions. These policies might help you [assess compliance](#) with the control. However, there often isn't a one-to-one or complete match between a control and one or more policies. As such, **Compliant** in Azure Policy refers only to the policies themselves. This doesn't ensure that you're fully compliant with all requirements of a control. In addition, the compliance standard includes controls that aren't addressed by any Azure Policy definitions at this time. Therefore, compliance in Azure Policy is only a partial view of your overall compliance status. The associations between controls and Azure Policy Regulatory Compliance definitions for these compliance standards can change over time.

## Australian Government ISM PROTECTED

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Australian Government ISM PROTECTED](#). For more information about this compliance standard, see [Australian Government ISM PROTECTED](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Guidelines for Personnel Security - Access to systems and their resources	415	User identification - 415	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	415	User identification - 415	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for Personnel Security - Access to systems and their resources	415	User identification - 415	<a href="#">Audit Windows machines that have the specified members in the Administrators group</a>	2.0.0
Guidelines for Personnel Security - Access to systems and their resources	415	User identification - 415	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Guidelines for System Hardening - Authentication hardening	421	Single-factor authentication - 421	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Guidelines for System Hardening - Authentication hardening	421	Single-factor authentication - 421	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Guidelines for System Hardening - Authentication hardening	421	Single-factor authentication - 421	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Guidelines for System Hardening - Authentication hardening	421	Single-factor authentication - 421	<a href="#">Windows machines should meet requirements for 'Security Settings - Account Policies'</a>	3.0.0
Guidelines for Personnel Security - Access to systems and their resources	445	Privileged access to systems - 445	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	445	Privileged access to systems - 445	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for Personnel Security - Access to systems and their resources	445	Privileged access to systems - 445	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Guidelines for Personnel Security - Access to systems and their resources	445	Privileged access to systems - 445	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Guidelines for Cryptography - Cryptographic fundamentals	459	Encrypting data at rest - 459	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Guidelines for System Monitoring - Event logging and auditing	582	Events to be logged - 582	Virtual machines should be connected to a specified workspace	1.1.0
Guidelines for System Management - System patching	940	When to patch security vulnerabilities - 940	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	940	When to patch security vulnerabilities - 940	Vulnerabilities in container security configurations should be remediated	3.0.0
Guidelines for System Management - System patching	940	When to patch security vulnerabilities - 940	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Guidelines for System Management - System patching	940	When to patch security vulnerabilities - 940	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Guidelines for Cryptography - Transport Layer Security	1139	Using Transport Layer Security - 1139	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for Cryptography - Transport Layer Security	1139	Using Transport Layer Security - 1139	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Guidelines for Cryptography - Transport Layer Security	1139	Using Transport Layer Security - 1139	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Guidelines for Cryptography - Transport Layer Security	1139	Using Transport Layer Security - 1139	Windows web servers should be configured to use secure communication protocols	4.0.0
Guidelines for System Management - System patching	1144	When to patch security vulnerabilities - 1144	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	1144	When to patch security vulnerabilities - 1144	Vulnerabilities in container security configurations should be remediated	3.0.0
Guidelines for System Management - System patching	1144	When to patch security vulnerabilities - 1144	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Guidelines for System Management - System patching	1144	When to patch security vulnerabilities - 1144	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Guidelines for Networking - Network design and configuration	1182	Network access controls - 1182	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Guidelines for Networking - Network design and configuration	1182	Network access controls - 1182	Internet-facing virtual machines should be protected with network security groups	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for Database Systems - Database servers	1277	Communications between database servers and web servers - 1277	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Guidelines for Database Systems - Database servers	1277	Communications between database servers and web servers - 1277	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Guidelines for Database Systems - Database servers	1277	Communications between database servers and web servers - 1277	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Guidelines for Database Systems - Database servers	1277	Communications between database servers and web servers - 1277	Windows web servers should be configured to use secure communication protocols	4.0.0
Guidelines for Gateways - Content filtering	1288	Antivirus scanning - 1288	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Guidelines for Gateways - Content filtering	1288	Antivirus scanning - 1288	Microsoft IaaSAntimalware extension should be deployed on Windows servers	1.1.0
Guidelines for Gateways - Content filtering	1288	Antivirus scanning - 1288	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Guidelines for System Management - System administration	1386	Restriction of management traffic flows - 1386	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Guidelines for System Hardening - Operating system hardening	1407	Operating system versions - 1407	System updates on virtual machine scale sets should be installed	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for System Hardening - Operating system hardening	1407	Operating system versions - 1407	System updates should be installed on your machines	4.0.0
Guidelines for System Hardening - Operating system hardening	1417	Antivirus software - 1417	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Guidelines for System Hardening - Operating system hardening	1417	Antivirus software - 1417	Microsoft IaaSAntimalware extension should be deployed on Windows servers	1.1.0
Guidelines for System Hardening - Operating system hardening	1417	Antivirus software - 1417	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Guidelines for Database Systems - Database servers	1425	Protecting database server contents - 1425	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Guidelines for System Management - System patching	1472	When to patch security vulnerabilities - 1472	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	1472	When to patch security vulnerabilities - 1472	Vulnerabilities in container security configurations should be remediated	3.0.0
Guidelines for System Management - System patching	1472	When to patch security vulnerabilities - 1472	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Guidelines for System Management - System patching	1472	When to patch security vulnerabilities - 1472	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Guidelines for System Hardening - Operating system hardening	1490	Application control - 1490	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for System Management - System patching	1494	When to patch security vulnerabilities - 1494	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	1494	When to patch security vulnerabilities - 1494	Vulnerabilities in container security configurations should be remediated	3.0.0
Guidelines for System Management - System patching	1494	When to patch security vulnerabilities - 1494	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Guidelines for System Management - System patching	1494	When to patch security vulnerabilities - 1494	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Guidelines for System Management - System patching	1495	When to patch security vulnerabilities - 1495	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	1495	When to patch security vulnerabilities - 1495	Vulnerabilities in container security configurations should be remediated	3.0.0
Guidelines for System Management - System patching	1495	When to patch security vulnerabilities - 1495	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Guidelines for System Management - System patching	1495	When to patch security vulnerabilities - 1495	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Guidelines for System Management - System patching	1496	When to patch security vulnerabilities - 1496	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Guidelines for System Management - System patching	1496	When to patch security vulnerabilities - 1496	Vulnerabilities in container security configurations should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for System Management - System patching	1496	When to patch security vulnerabilities - 1496	<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	3.0.0
Guidelines for System Management - System patching	1496	When to patch security vulnerabilities - 1496	<a href="#">Vulnerabilities in security configuration on your virtual machine scale sets should be remediated</a>	3.0.0
Guidelines for Personnel Security - Access to systems and their resources	1503	Standard access to systems - 1503	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	1503	Standard access to systems - 1503	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	1503	Standard access to systems - 1503	<a href="#">Audit Windows machines that have the specified members in the Administrators group</a>	2.0.0
Guidelines for Personnel Security - Access to systems and their resources	1503	Standard access to systems - 1503	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Guidelines for Personnel Security - Access to systems and their resources	1507	Privileged access to systems - 1507	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	1507	Privileged access to systems - 1507	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for Personnel Security - Access to systems and their resources	1507	Privileged access to systems - 1507	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Guidelines for Personnel Security - Access to systems and their resources	1507	Privileged access to systems - 1507	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Guidelines for Personnel Security - Access to systems and their resources	1508	Privileged access to systems - 1508	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	1508	Privileged access to systems - 1508	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Guidelines for Personnel Security - Access to systems and their resources	1508	Privileged access to systems - 1508	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Guidelines for Personnel Security - Access to systems and their resources	1508	Privileged access to systems - 1508	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Guidelines for Personnel Security - Access to systems and their resources	1508	Privileged access to systems - 1508	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Guidelines for System Management - Data backup and restoration	1511	Performing backups - 1511	Audit virtual machines without disaster recovery configured	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Guidelines for System Hardening - Authentication hardening	1546	Authenticating to systems - 1546	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Guidelines for System Hardening - Authentication hardening	1546	Authenticating to systems - 1546	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Guidelines for System Hardening - Authentication hardening	1546	Authenticating to systems - 1546	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Guidelines for System Hardening - Authentication hardening	1546	Authenticating to systems - 1546	Audit Linux machines that have accounts without passwords	3.0.0
Guidelines for System Hardening - Authentication hardening	1546	Authenticating to systems - 1546	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0

## Azure Security Benchmark

The [Azure Security Benchmark](#) provides recommendations on how you can secure your cloud solutions on Azure. To see how this service completely maps to the Azure Security Benchmark, see the [Azure Security Benchmark mapping files](#).

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Azure Security Benchmark](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Network Security	NS-1	Establish network segmentation boundaries	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Security	NS-1	Establish network segmentation boundaries	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Network Security	NS-1	Establish network segmentation boundaries	Internet-facing virtual machines should be protected with network security groups	3.0.0
Network Security	NS-1	Establish network segmentation boundaries	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Network Security	NS-3	Deploy firewall at the edge of enterprise network	IP Forwarding on your virtual machine should be disabled	3.0.0
Network Security	NS-3	Deploy firewall at the edge of enterprise network	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Network Security	NS-3	Deploy firewall at the edge of enterprise network	Management ports should be closed on your virtual machines	3.0.0
Network Security	NS-7	Simplify network security configuration	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Identity Management	IM-3	Manage application identities securely and automatically	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Identity Management	IM-6	Use strong authentication controls	Authentication to Linux machines should require SSH keys	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Privileged Access	PA-2	Avoid standing access for accounts and permissions	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Data Protection	DP-3	Encrypt sensitive data in transit	Windows web servers should be configured to use secure communication protocols	4.0.0
Data Protection	DP-4	Enable data at rest encryption by default	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Asset Management	AM-2	Use only approved services		
Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0			
Asset Management	AM-5	Use only approved applications in virtual machine	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Asset Management	AM-5	Use only approved applications in virtual machine	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Logging and Threat Detection	LT-1	Enable threat detection capabilities	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
Logging and Threat Detection	LT-2	Enable threat detection for identity and access management	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
Logging and Threat Detection	LT-3	Enable logging for security investigation	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0

Domain	Control ID	Control Title	Policy	Policy Version
Logging and Threat Detection	LT-4	Enable network logging for security investigation		
[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview			
Logging and Threat Detection	LT-4	Enable network logging for security investigation		
[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview			
Logging and Threat Detection	LT-5	Centralize security log management and analysis	<a href="#">Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring</a>	1.0.0
Logging and Threat Detection	LT-5	Centralize security log management and analysis	<a href="#">Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring</a>	1.0.0
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	<a href="#">[Preview]: Guest Attestation extension should be installed on supported Linux virtual machines</a>	5.0.0-preview
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	<a href="#">[Preview]: Guest Attestation extension should be installed on supported Linux virtual machines scale sets</a>	4.0.0-preview
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	<a href="#">[Preview]: Guest Attestation extension should be installed on supported Windows virtual machines</a>	3.0.0-preview

Domain	Control ID	Control Title	Policy	Policy Version
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	[Preview]: Guest Attestation extension should be installed on supported Windows virtual machines scale sets	2.0.0-preview
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	[Preview]: Secure Boot should be enabled on supported Windows virtual machines	3.0.0-preview
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	[Preview]: vTPM should be enabled on supported virtual machines	2.0.0-preview
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	Guest Configuration extension should be installed on your machines	1.0.2
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	Linux machines should meet requirements for the Azure compute security baseline	2.0.0
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Posture and Vulnerability Management	PV-4	Audit and enforce secure configurations for compute resources	Windows machines should meet requirements of the Azure compute security baseline	2.0.0
Posture and Vulnerability Management	PV-5	Perform vulnerability assessments	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	[Preview]: Machines should be configured to periodically check for missing system updates	1.0.0-preview

Domain	Control ID	Control Title	Policy	Policy Version
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	[Preview]: System updates should be installed on your machines (powered by Update Center)	1.0.0-preview
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	SQL servers on machines should have vulnerability findings resolved	1.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	System updates on virtual machine scale sets should be installed	3.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	System updates should be installed on your machines	4.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	Vulnerabilities in container security configurations should be remediated	3.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Posture and Vulnerability Management	PV-6	Rapidly and automatically remediate vulnerabilities	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Endpoint Security	ES-2	Use modern anti-malware software	Endpoint protection health issues should be resolved on your machines	1.0.0
Endpoint Security	ES-2	Use modern anti-malware software	Endpoint protection should be installed on your machines	1.0.0
Endpoint Security	ES-2	Use modern anti-malware software	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Endpoint Security	ES-2	Use modern anti-malware software	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Endpoint Security	ES-2	Use modern anti-malware software	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
Endpoint Security	ES-3	Ensure anti-malware software and signatures are updated	Endpoint protection health issues should be resolved on your machines	1.0.0
Backup and Recovery	BR-1	Ensure regular automated backups	Azure Backup should be enabled for Virtual Machines	3.0.0
Backup and Recovery	BR-2	Protect backup and recovery data	Azure Backup should be enabled for Virtual Machines	3.0.0
DevOps Security	DS-6	Enforce security of workload throughout DevOps lifecycle	Vulnerabilities in container security configurations should be remediated	3.0.0

## Azure Security Benchmark v1

The [Azure Security Benchmark](#) provides recommendations on how you can secure your cloud solutions on Azure. To see how this service completely maps to the Azure Security Benchmark, see the [Azure Security Benchmark mapping files](#).

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Azure Security Benchmark](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Network Security	1.1	Protect resources using Network Security Groups or Azure Firewall on your Virtual Network	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Network Security	1.1	Protect resources using Network Security Groups or Azure Firewall on your Virtual Network	Internet-facing virtual machines should be protected with network security groups	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Security	1.1	Protect resources using Network Security Groups or Azure Firewall on your Virtual Network	IP Forwarding on your virtual machine should be disabled	3.0.0
Network Security	1.1	Protect resources using Network Security Groups or Azure Firewall on your Virtual Network	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Network Security	1.1	Protect resources using Network Security Groups or Azure Firewall on your Virtual Network	Management ports should be closed on your virtual machines	3.0.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Windows machines should meet requirements for 'Administrative Templates - Network'	3.0.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Windows machines should meet requirements for 'Security Options - Microsoft Network Server'	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Windows machines should meet requirements for 'Security Options - Network Access'	3.0.0
Network Security	1.11	Use automated tools to monitor network resource configurations and detect changes	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
Network Security	1.4	Deny communications with known malicious IP addresses	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Network Security	1.4	Deny communications with known malicious IP addresses	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Logging and Monitoring	2.2	Configure central security log management	Audit Windows machines on which the Log Analytics agent is not connected as expected	2.0.0
Logging and Monitoring	2.2	Configure central security log management	The Log Analytics extension should be installed on Virtual Machine Scale Sets	1.0.1
Logging and Monitoring	2.2	Configure central security log management	Virtual machines should have the Log Analytics extension installed	1.0.1
Logging and Monitoring	2.3	Enable audit logging for Azure resources	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Logging and Monitoring	2.4	Collect security logs from operating systems	Audit Windows machines on which the Log Analytics agent is not connected as expected	2.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Logging and Monitoring	2.4	Collect security logs from operating systems	The Log Analytics extension should be installed on Virtual Machine Scale Sets	1.0.1
Logging and Monitoring	2.4	Collect security logs from operating systems	Virtual machines should have the Log Analytics extension installed	1.0.1
Logging and Monitoring	2.8	Centralize anti-malware logging	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Logging and Monitoring	2.8	Centralize anti-malware logging	Microsoft Antimalware for Azure should be configured to automatically update protection signatures	1.0.0
Logging and Monitoring	2.8	Centralize anti-malware logging	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Identity and Access Control	3.3	Use dedicated administrative accounts	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Identity and Access Control	3.3	Use dedicated administrative accounts	Audit Windows machines that have extra accounts in the Administrators group	2.0.0
Identity and Access Control	3.3	Use dedicated administrative accounts	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Data Protection	4.8	Encrypt sensitive information at rest	[Deprecated]: Unattached disks should be encrypted	1.0.0-deprecated
Data Protection	4.8	Encrypt sensitive information at rest	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3

Domain	Control ID	Control Title	Policy	Policy Version
Vulnerability Management	5.1	Run automated vulnerability scanning tools	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Vulnerability Management	5.2	Deploy automated operating system patch management solution	System updates on virtual machine scale sets should be installed	3.0.0
Vulnerability Management	5.2	Deploy automated operating system patch management solution	System updates should be installed on your machines	4.0.0
Vulnerability Management	5.5	Use a risk-rating process to prioritize the remediation of discovered vulnerabilities	Vulnerabilities in container security configurations should be remediated	3.0.0
Vulnerability Management	5.5	Use a risk-rating process to prioritize the remediation of discovered vulnerabilities	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Vulnerability Management	5.5	Use a risk-rating process to prioritize the remediation of discovered vulnerabilities	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Inventory and Asset Management	6.10	Implement approved application list	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Inventory and Asset Management	6.8	Use only approved applications	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Inventory and Asset Management	6.9	Use only approved Azure services	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Secure Configuration	7.10	Implement automated configuration monitoring for operating systems	Vulnerabilities in container security configurations should be remediated	3.0.0
Secure Configuration	7.10	Implement automated configuration monitoring for operating systems	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Secure Configuration	7.10	Implement automated configuration monitoring for operating systems	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Secure Configuration	7.4	Maintain secure operating system configurations	Vulnerabilities in container security configurations should be remediated	3.0.0
Secure Configuration	7.4	Maintain secure operating system configurations	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Secure Configuration	7.4	Maintain secure operating system configurations	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Malware Defense	8.1	Use centrally managed anti-malware software	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Malware Defense	8.1	Use centrally managed anti-malware software	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Malware Defense	8.3	Ensure anti-malware software and signatures are updated	Microsoft Antimalware for Azure should be configured to automatically update protection signatures	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Data Recovery	9.1	Ensure regular automated back ups	Azure Backup should be enabled for Virtual Machines	3.0.0
Data Recovery	9.2	Perform complete system backups and backup any customer managed keys	Azure Backup should be enabled for Virtual Machines	3.0.0

## Canada Federal PBMM

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Canada Federal PBMM](#). For more information about this compliance standard, see [Canada Federal PBMM](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-5	Separation of Duties	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-5	Separation of Duties	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-5	Separation of Duties	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Access Control	AC-5	Separation of Duties	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Access Control	AC-5	Separation of Duties	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC-6	Least Privilege	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-6	Least Privilege	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-6	Least Privilege	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Access Control	AC-6	Least Privilege	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Access Control	AC-6	Least Privilege	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17(1)	Remote Access   Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17(1)	Remote Access   Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17(1)	Remote Access   Automated Monitoring / Control	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-17(1)	Remote Access   Automated Monitoring / Control	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Audit and Accountability	AU-3	Content of Audit Records	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Audit and Accountability	AU-3	Content of Audit Records	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Audit and Accountability	AU-3	Content of Audit Records	Virtual machines should be connected to a specified workspace	1.1.0
Audit and Accountability	AU-12	Audit Generation	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Audit and Accountability	AU-12	Audit Generation	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Audit and Accountability	AU-12	Audit Generation	Virtual machines should be connected to a specified workspace	1.1.0
Configuration Management	CM-7(5)	Least Functionality   Authorized Software / Whitelisting	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-11	User-Installed Software	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Contingency Planning	CP-7	Alternative Processing Site	Audit virtual machines without disaster recovery configured	1.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Linux machines that have accounts without passwords	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Audit Windows machines that do not have the password complexity setting enabled	2.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identification and Authentication	IA-5(1)	Authenticator Management   Password-Based Authentication	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Risk Assessment	RA-5	Vulnerability Scanning	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-7	Boundary Protection	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7(3)	Boundary Protection   Access Points	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7(4)	Boundary Protection   External Telecommunications Services	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-8(1)	Transmission Confidentiality and Integrity   Cryptographic or Alternate Physical Protection	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Information Integrity	SI-2	Flaw Remediation	System updates on virtual machine scale sets should be installed	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates should be installed on your machines	4.0.0
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your machines should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3(1)	Malicious Code Protection   Central Management	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3(1)	Malicious Code Protection   Central Management	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-4	Information System Monitoring	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
System and Information Integrity	SI-4	Information System Monitoring	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
System and Information Integrity	SI-4	Information System Monitoring	Virtual machines should be connected to a specified workspace	1.1.0

## CIS Microsoft Azure Foundations Benchmark 1.1.0

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - CIS Microsoft Azure Foundations Benchmark 1.1.0](#). For more information about this compliance standard, see [CIS Microsoft Azure Foundations Benchmark](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
--------	------------	---------------	--------------------------	----------------------------

Domain	Control ID	Control Title	Policy	Policy Version
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.10	Ensure ASC Default policy setting "Monitor Vulnerability Assessment" is not "Disabled"	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.12	Ensure ASC Default policy setting "Monitor JIT Network Access" is not "Disabled"	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.13	Ensure ASC Default policy setting "Monitor Adaptive Application Whitelisting" is not "Disabled"	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.3	Ensure ASC Default policy setting "Monitor System Updates" is not "Disabled"	System updates should be installed on your machines	4.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.4	Ensure ASC Default policy setting "Monitor OS Vulnerabilities" is not "Disabled"	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.5	Ensure ASC Default policy setting "Monitor Endpoint Protection" is not "Disabled"	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.6	Ensure ASC Default policy setting "Monitor Disk Encryption" is not "Disabled"	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.7	Ensure ASC Default policy setting "Monitor Network Security Groups" is not "Disabled"	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Security Center	CIS Microsoft Azure Foundations Benchmark recommendation 2.9	Ensure ASC Default policy setting "Enable Next Generation Firewall(NGFW) Monitoring" is not "Disabled"	Internet-facing virtual machines should be protected with network security groups	3.0.0
Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.1	Ensure that 'OS disk' are encrypted	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.2	Ensure that 'Data disks' are encrypted	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.4	Ensure that only approved extensions are installed	Only approved VM extensions should be installed	1.0.0
Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.5	Ensure that the latest OS Patches for all Virtual Machines are applied	System updates should be installed on your machines	4.0.0
Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.6	Ensure that the endpoint protection for all Virtual Machines is installed	Monitor missing Endpoint Protection in Azure Security Center	3.0.0

## CIS Microsoft Azure Foundations Benchmark 1.3.0

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - CIS Microsoft Azure Foundations Benchmark 1.3.0](#). For more information about this compliance standard, see [CIS Microsoft Azure Foundations Benchmark](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
5 Logging and Monitoring	CIS Microsoft Azure Foundations Benchmark recommendation 5.3	Ensure that Diagnostic Logs are enabled for all services which support it.	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
7 Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.1	Ensure Virtual Machines are utilizing Managed Disks	Audit VMs that do not use managed disks	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
7 Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.2	Ensure that 'OS and Data' disks are encrypted with CMK	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
7 Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.4	Ensure that only approved extensions are installed	Only approved VM extensions should be installed	1.0.0
7 Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.5	Ensure that the latest OS Patches for all Virtual Machines are applied	System updates should be installed on your machines	4.0.0
7 Virtual Machines	CIS Microsoft Azure Foundations Benchmark recommendation 7.6	Ensure that the endpoint protection for all Virtual Machines is installed	Monitor missing Endpoint Protection in Azure Security Center	3.0.0

## CMMC Level 3

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - CMMC Level 3](#). For more information about this compliance standard, see [Cybersecurity Maturity Model Certification \(CMMC\)](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	<a href="#">Audit Linux machines that allow remote connections from accounts without passwords</a>	3.0.0
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	<a href="#">Management ports of virtual machines should be protected with just-in-time network access control</a>	3.0.0
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	<a href="#">Windows machines should meet requirements for 'Security Options - Network Access'</a>	3.0.0
Access Control	AC.1.001	Limit information system access to authorized users, processes acting on behalf of authorized users, and devices (including other information systems).	<a href="#">Windows machines should meet requirements for 'Security Options - Network Security'</a>	3.0.0
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	<a href="#">Audit Linux machines that allow remote connections from accounts without passwords</a>	3.0.0
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	<a href="#">Management ports of virtual machines should be protected with just-in-time network access control</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	Windows machines should meet requirements for 'Security Options - Network Access'	3.0.0
Access Control	AC.1.002	Limit information system access to the types of transactions and functions that authorized users are permitted to execute.	Windows web servers should be configured to use secure communication protocols	4.0.0
Access Control	AC.1.003	Verify and control/limit connections to and use of external information systems.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Access Control	AC.1.003	Verify and control/limit connections to and use of external information systems.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC.2.007	Employ the principle of least privilege, including for specific security functions and privileged accounts.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC.2.008	Use non-privileged accounts or roles when accessing nonsecurity functions.	Windows machines should meet requirements for 'Security Options - User Account Control'	3.0.0
Access Control	AC.2.008	Use non-privileged accounts or roles when accessing nonsecurity functions.	Windows machines should meet requirements for 'User Rights Assignment'	3.0.0
Access Control	AC.2.013	Monitor and control remote access sessions.	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC.2.013	Monitor and control remote access sessions.	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC.2.013	Monitor and control remote access sessions.	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC.2.013	Monitor and control remote access sessions.	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC.2.013	Monitor and control remote access sessions.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC.2.013	Monitor and control remote access sessions.	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
Access Control	AC.2.016	Control the flow of CUI in accordance with approved authorizations.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Access Control	AC.2.016	Control the flow of CUI in accordance with approved authorizations.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC.2.016	Control the flow of CUI in accordance with approved authorizations.	Windows machines should meet requirements for 'Security Options - Network Access'	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC.3.017	Separate the duties of individuals to reduce the risk of malevolent activity without collusion.	<a href="#">Audit Windows machines missing any of specified members in the Administrators group</a>	2.0.0
Access Control	AC.3.017	Separate the duties of individuals to reduce the risk of malevolent activity without collusion.	<a href="#">Audit Windows machines that have the specified members in the Administrators group</a>	2.0.0
Access Control	AC.3.018	Prevent non-privileged users from executing privileged functions and capture the execution of such functions in audit logs.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Privilege Use'</a>	3.0.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs</a>	3.0.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Guest Configuration extension should be installed on your machines</a>	1.0.2

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity</a>	1.0.1
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Windows machines should meet requirements for 'Security Options - User Account Control'</a>	3.0.0
Access Control	AC.3.021	Authorize remote execution of privileged commands and remote access to security-relevant information.	<a href="#">Windows machines should meet requirements for 'User Rights Assignment'</a>	3.0.0
Audit and Accountability	AU.2.041	Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.	<a href="#">[Preview]: Log Analytics Extension should be enabled for listed virtual machine images</a>	2.0.1-preview
Audit and Accountability	AU.2.041	Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.	<a href="#">Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images</a>	2.0.1
Audit and Accountability	AU.2.041	Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.	<a href="#">The Log Analytics extension should be installed on Virtual Machine Scale Sets</a>	1.0.1
Audit and Accountability	AU.2.041	Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.	<a href="#">Virtual machines should be connected to a specified workspace</a>	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Audit and Accountability	AU.2.041	Ensure that the actions of individual system users can be uniquely traced to those users so they can be held accountable for their actions.	<a href="#">Virtual machines should have the Log Analytics extension installed</a>	1.0.1
Audit and Accountability	AU.2.042	Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.	<a href="#">[Preview]: Log Analytics Extension should be enabled for listed virtual machine images</a>	2.0.1-preview
Audit and Accountability	AU.2.042	Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.	<a href="#">Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images</a>	2.0.1
Audit and Accountability	AU.2.042	Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.	<a href="#">The Log Analytics extension should be installed on Virtual Machine Scale Sets</a>	1.0.1
Audit and Accountability	AU.2.042	Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.	<a href="#">Virtual machines should be connected to a specified workspace</a>	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Audit and Accountability	AU.2.042	Create and retain system audit logs and records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful or unauthorized system activity.	<a href="#">Virtual machines should have the Log Analytics extension installed</a>	1.0.1
Audit and Accountability	AU.3.046	Alert in the event of an audit logging process failure.	<a href="#">[Preview]: Log Analytics Extension should be enabled for listed virtual machine images</a>	2.0.1-preview
Audit and Accountability	AU.3.046	Alert in the event of an audit logging process failure.	<a href="#">Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images</a>	2.0.1
Audit and Accountability	AU.3.046	Alert in the event of an audit logging process failure.	<a href="#">Virtual machines should be connected to a specified workspace</a>	1.1.0
Audit and Accountability	AU.3.048	Collect audit information (e.g., logs) into one or more central repositories.	<a href="#">[Preview]: Log Analytics Extension should be enabled for listed virtual machine images</a>	2.0.1-preview
Audit and Accountability	AU.3.048	Collect audit information (e.g., logs) into one or more central repositories.	<a href="#">Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images</a>	2.0.1
Audit and Accountability	AU.3.048	Collect audit information (e.g., logs) into one or more central repositories.	<a href="#">The Log Analytics extension should be installed on Virtual Machine Scale Sets</a>	1.0.1
Audit and Accountability	AU.3.048	Collect audit information (e.g., logs) into one or more central repositories.	<a href="#">Virtual machines should be connected to a specified workspace</a>	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Audit and Accountability	AU.3.048	Collect audit information (e.g., logs) into one or more central repositories.	<a href="#">Virtual machines should have the Log Analytics extension installed</a>	1.0.1
Security Assessment	CA.2.158	Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.	<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	3.0.0
Security Assessment	CA.2.158	Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Security Assessment	CA.2.158	Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.	<a href="#">Allowlist rules in your adaptive application control policy should be updated</a>	3.0.0
Security Assessment	CA.2.158	Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.	<a href="#">Endpoint protection solution should be installed on virtual machine scale sets</a>	3.0.0
Security Assessment	CA.2.158	Periodically assess the security controls in organizational systems to determine if the controls are effective in their application.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
Security Assessment	CA.3.161	Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.	<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Security Assessment	CA.3.161	Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Security Assessment	CA.3.161	Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.	<a href="#">Allowlist rules in your adaptive application control policy should be updated</a>	3.0.0
Security Assessment	CA.3.161	Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.	<a href="#">Endpoint protection solution should be installed on virtual machine scale sets</a>	3.0.0
Security Assessment	CA.3.161	Monitor security controls on an ongoing basis to ensure the continued effectiveness of the controls.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
Configuration Management	CM.2.061	Establish and maintain baseline configurations and inventories of organizational systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Configuration Management	CM.2.061	Establish and maintain baseline configurations and inventories of organizational systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles.	<a href="#">Linux machines should meet requirements for the Azure compute security baseline</a>	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Configuration Management	CM.2.062	Employ the principle of least functionality by configuring organizational systems to provide only essential capabilities.	Windows machines should meet requirements for 'System Audit Policies - Privilege Use'	3.0.0
Configuration Management	CM.2.063	Control and monitor user-installed software.	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM.2.063	Control and monitor user-installed software.	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM.2.063	Control and monitor user-installed software.	Windows machines should meet requirements for 'Security Options - User Account Control'	3.0.0
Configuration Management	CM.2.064	Establish and enforce security configuration settings for information technology products employed in organizational systems.	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Configuration Management	CM.2.064	Establish and enforce security configuration settings for information technology products employed in organizational systems.	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
Configuration Management	CM.2.065	Track, review, approve or disapprove, and log changes to organizational systems.	Windows machines should meet requirements for 'System Audit Policies - Policy Change'	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Configuration Management	CM.3.068	Restrict, disable, or prevent the use of nonessential programs, functions, ports, protocols, and services.	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Configuration Management	CM.3.069	Apply deny-by-exception (blacklisting) policy to prevent the use of unauthorized software or deny-all, permit-by-exception (whitelisting) policy to allow the execution of authorized software.	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Audit Linux machines that do not have the passwd file permissions set to 0644</a>	3.0.0
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Audit Linux machines that have accounts without passwords</a>	3.0.0
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Identification and Authentication	IA.1.077	Authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	<a href="#">Windows machines should meet requirements for 'Security Options - Network Security'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Audit Linux machines that have accounts without passwords</a>	3.0.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Audit Windows machines that do not have the password complexity setting enabled</a>	2.0.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Audit Windows machines that do not restrict the minimum password length to 14 characters</a>	2.0.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Identification and Authentication	IA.2.078	Enforce a minimum password complexity and change of characters when new passwords are created.	<a href="#">Windows machines should meet requirements for 'Security Options - Network Security'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA.2.079	Prohibit password reuse for a specified number of generations.	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA.2.079	Prohibit password reuse for a specified number of generations.	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA.2.079	Prohibit password reuse for a specified number of generations.	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	IA.2.079	Prohibit password reuse for a specified number of generations.	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Identification and Authentication	IA.2.079	Prohibit password reuse for a specified number of generations.	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
Identification and Authentication	IA.2.081	Store and transmit only cryptographically-protected passwords.	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA.2.081	Store and transmit only cryptographically-protected passwords.	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA.2.081	Store and transmit only cryptographically-protected passwords.	Audit Windows machines that do not store passwords using reversible encryption	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA.2.081	Store and transmit only cryptographically-protected passwords.	<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	1.2.0
Identification and Authentication	IA.2.081	Store and transmit only cryptographically-protected passwords.	<a href="#">Windows machines should meet requirements for 'Security Options - Network Security'</a>	3.0.0
Identification and Authentication	IA.3.084	Employ replay-resistant authentication mechanisms for network access to privileged and nonprivileged accounts.	<a href="#">Windows web servers should be configured to use secure communication protocols</a>	4.0.0
Incident Response	IR.2.093	Detect and report events.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
Recovery	RE.2.137	Regularly perform and test data backups.	<a href="#">Audit virtual machines without disaster recovery configured</a>	1.0.0
Recovery	RE.2.137	Regularly perform and test data backups.	<a href="#">Azure Backup should be enabled for Virtual Machines</a>	3.0.0
Recovery	RE.3.139	Regularly perform complete, comprehensive and resilient data backups as organizationally-defined.	<a href="#">Audit virtual machines without disaster recovery configured</a>	1.0.0
Recovery	RE.3.139	Regularly perform complete, comprehensive and resilient data backups as organizationally-defined.	<a href="#">Azure Backup should be enabled for Virtual Machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Risk Assessment	RM.2.141	Periodically assess the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, and individuals, resulting from the operation of organizational systems and the associated processing, storage, or transmission of CUI.	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RM.2.142	Scan for vulnerabilities in organizational systems and applications periodically and when new vulnerabilities affecting those systems and applications are identified.	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RM.2.143	Remediate vulnerabilities in accordance with risk assessments.	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RM.2.143	Remediate vulnerabilities in accordance with risk assessments.	Vulnerabilities in container security configurations should be remediated	3.0.0
Risk Assessment	RM.2.143	Remediate vulnerabilities in accordance with risk assessments.	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	RM.2.143	Remediate vulnerabilities in accordance with risk assessments.	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	<a href="#">All network ports should be restricted on network security groups associated to your virtual machine</a>	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	<a href="#">Management ports of virtual machines should be protected with just-in-time network access control</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Windows machines should meet requirements for 'Security Options - Network Access'	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
System and Communications Protection	SC.1.175	Monitor, control, and protect communications (i.e., information transmitted or received by organizational systems) at the external boundaries and key internal boundaries of organizational systems.	Windows web servers should be configured to use secure communication protocols	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC.1.176	Implement subnetworks for publicly accessible system components that are physically or logically separated from internal networks.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC.1.176	Implement subnetworks for publicly accessible system components that are physically or logically separated from internal networks.	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC.1.176	Implement subnetworks for publicly accessible system components that are physically or logically separated from internal networks.	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC.2.179	Use encrypted sessions for the management of network devices.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC.3.177	Employ FIPS-validated cryptography when used to protect the confidentiality of CUI.	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
System and Communications Protection	SC.3.177	Employ FIPS-validated cryptography when used to protect the confidentiality of CUI.	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Communications Protection	SC.3.181	Separate user functionality from system management functionality.	Audit Windows machines that have the specified members in the Administrators group	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Windows machines should meet requirements for 'Security Options - Network Access'	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC.3.183	Deny network communications traffic by default and allow network communications traffic by exception (i.e., deny all, permit by exception).	Windows machines should meet requirements for 'Security Options - Network Security'	3.0.0
System and Communications Protection	SC.3.185	Implement cryptographic mechanisms to prevent unauthorized disclosure of CUI during transmission unless otherwise protected by alternative physical safeguards.	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC.3.190	Protect the authenticity of communications sessions.	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC.3.191	Protect the confidentiality of CUI at rest.	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Information Integrity	SI.1.210	Identify, report, and correct information and information system flaws in a timely manner.	Microsoft Antimalware for Azure should be configured to automatically update protection signatures	1.0.0
System and Information Integrity	SI.1.210	Identify, report, and correct information and information system flaws in a timely manner.	System updates on virtual machine scale sets should be installed	3.0.0
System and Information Integrity	SI.1.210	Identify, report, and correct information and information system flaws in a timely manner.	System updates should be installed on your machines	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI.1.210	Identify, report, and correct information and information system flaws in a timely manner.	<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	3.0.0
System and Information Integrity	SI.1.210	Identify, report, and correct information and information system flaws in a timely manner.	<a href="#">Vulnerabilities in security configuration on your virtual machine scale sets should be remediated</a>	3.0.0
System and Information Integrity	SI.1.211	Provide protection from malicious code at appropriate locations within organizational information systems.	<a href="#">Endpoint protection solution should be installed on virtual machine scale sets</a>	3.0.0
System and Information Integrity	SI.1.211	Provide protection from malicious code at appropriate locations within organizational information systems.	<a href="#">Microsoft Antimalware for Azure should be configured to automatically update protection signatures</a>	1.0.0
System and Information Integrity	SI.1.211	Provide protection from malicious code at appropriate locations within organizational information systems.	<a href="#">Microsoft IaaSAntimalware extension should be deployed on Windows servers</a>	1.1.0
System and Information Integrity	SI.1.211	Provide protection from malicious code at appropriate locations within organizational information systems.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
System and Information Integrity	SI.1.212	Update malicious code protection mechanisms when new releases are available.	<a href="#">Microsoft Antimalware for Azure should be configured to automatically update protection signatures</a>	1.0.0
System and Information Integrity	SI.1.213	Perform periodic scans of the information system and real-time scans of files from external sources as files are downloaded, opened, or executed.	<a href="#">Microsoft Antimalware for Azure should be configured to automatically update protection signatures</a>	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI.1.213	Perform periodic scans of the information system and real-time scans of files from external sources as files are downloaded, opened, or executed.	<a href="#">Microsoft IaaSAntimalware extension should be deployed on Windows servers</a>	1.1.0
System and Information Integrity	SI.1.213	Perform periodic scans of the information system and real-time scans of files from external sources as files are downloaded, opened, or executed.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0

## FedRAMP High

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP High](#). For more information about this compliance standard, see [FedRAMP High](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-2 (12)	Account Monitoring / Atypical Usage	<a href="#">Management ports of virtual machines should be protected with just-in-time network access control</a>	3.0.0
Access Control	AC-3	Access Enforcement	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access Control	AC-3	Access Enforcement	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Access Control	AC-3	Access Enforcement	<a href="#">Audit Linux machines that have accounts without passwords</a>	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-3	Access Enforcement	Authentication to Linux machines should require SSH keys	3.0.0
Access Control	AC-3	Access Enforcement	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-3	Access Enforcement	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Access Control	AC-4	Information Flow Enforcement	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Access Control	AC-4	Information Flow Enforcement	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Access Control	AC-4	Information Flow Enforcement	Disk access resources should use private link	1.0.0
Access Control	AC-4	Information Flow Enforcement	Internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-4	Information Flow Enforcement	IP Forwarding on your virtual machine should be disabled	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports should be closed on your virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC-4	Information Flow Enforcement	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17	Remote Access	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17	Remote Access	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-17	Remote Access	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17	Remote Access	Disk access resources should use private link	1.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC-17 (1)	Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Disk access resources should use private link	1.0.0
Audit and Accountability	AU-6	Audit Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6	Audit Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (4)	Central Review and Analysis	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (4)	Central Review and Analysis	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-6 (5)	Integration / Scanning and Monitoring Capabilities	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Audit and Accountability	AU-12	Audit Generation	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Generation	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Generation	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-12	Audit Generation	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Generation	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Generation	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-12	Audit Generation	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-12 (1)	System-wide / Time-correlated Audit Trail	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Configuration Management	CM-6	Configuration Settings	Linux machines should meet requirements for the Azure compute security baseline	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Configuration Management	CM-6	Configuration Settings	Windows machines should meet requirements of the Azure compute security baseline	2.0.0
Configuration Management	CM-7	Least Functionality	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7	Least Functionality	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (5)	Authorized Software / Whitelisting	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (5)	Authorized Software / Whitelisting	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-11	User-installed Software	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Configuration Management	CM-11	User-installed Software	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Contingency Planning	CP-7	Alternate Processing Site	Audit virtual machines without disaster recovery configured	1.0.0
Contingency Planning	CP-9	Information System Backup	Azure Backup should be enabled for Virtual Machines	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	IA-5	Authenticator Management	Authentication to Linux machines should require SSH keys	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA-5	Authenticator Management	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have the password complexity setting enabled	2.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Risk Assessment	RA-5	Vulnerability Scanning	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	SQL servers on machines should have vulnerability findings resolved	1.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in container security configurations should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Communications Protection	SC-3	Security Function Isolation	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Communications Protection	SC-3	Security Function Isolation	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Communications Protection	SC-3	Security Function Isolation	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Communications Protection	SC-5	Denial of Service Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7	Boundary Protection	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7	Boundary Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports should be closed on your virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-7	Boundary Protection	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7 (3)	Access Points	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports should be closed on your virtual machines	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-8	Transmission Confidentiality and Integrity	Windows web servers should be configured to use secure communication protocols	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Communications Protection	SC-8 (1)	Cryptographic or Alternate Physical Protection	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	Managed disks should be double encrypted with both platform-managed and customer-managed keys	1.0.0
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	OS and data disks should be encrypted with a customer-managed key	3.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Information Integrity	SI-2	Flaw Remediation	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates on virtual machine scale sets should be installed	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Information Integrity	SI-2	Flaw Remediation	System updates should be installed on your machines	4.0.0
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Information Integrity	SI-3 (1)	Central Management	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3 (1)	Central Management	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3 (1)	Central Management	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Information Integrity	SI-4	Information System Monitoring	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI-4	Information System Monitoring	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
System and Information Integrity	SI-4	Information System Monitoring	Guest Configuration extension should be installed on your machines	1.0.2
System and Information Integrity	SI-4	Information System Monitoring	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
System and Information Integrity	SI-4	Information System Monitoring	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
System and Information Integrity	SI-4	Information System Monitoring	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
System and Information Integrity	SI-16	Memory Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0

## FedRAMP Moderate

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - FedRAMP Moderate](#). For more information about this compliance standard, see [FedRAMP Moderate](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-2 (12)	Account Monitoring / Atypical Usage	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-3	Access Enforcement	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-3	Access Enforcement	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-3	Access Enforcement	Audit Linux machines that have accounts without passwords	3.0.0
Access Control	AC-3	Access Enforcement	Authentication to Linux machines should require SSH keys	3.0.0
Access Control	AC-3	Access Enforcement	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-3	Access Enforcement	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Access Control	AC-4	Information Flow Enforcement	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Access Control	AC-4	Information Flow Enforcement	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Access Control	AC-4	Information Flow Enforcement	Disk access resources should use private link	1.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-4	Information Flow Enforcement	Internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-4	Information Flow Enforcement	IP Forwarding on your virtual machine should be disabled	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports should be closed on your virtual machines	3.0.0
Access Control	AC-4	Information Flow Enforcement	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17	Remote Access	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17	Remote Access	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-17	Remote Access	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17	Remote Access	Disk access resources should use private link	1.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17 (1)	Automated Monitoring / Control	Disk access resources should use private link	1.0.0
Audit and Accountability	AU-6	Audit Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-6	Audit Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Generation	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Generation	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Generation	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-12	Audit Generation	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Generation	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Generation	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-12	Audit Generation	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Configuration Management	CM-6	Configuration Settings	Linux machines should meet requirements for the Azure compute security baseline	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Configuration Management	CM-6	Configuration Settings	Windows machines should meet requirements of the Azure compute security baseline	2.0.0
Configuration Management	CM-7	Least Functionality	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7	Least Functionality	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (5)	Authorized Software / Whitelisting	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (5)	Authorized Software / Whitelisting	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-11	User-installed Software	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Configuration Management	CM-11	User-installed Software	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Contingency Planning	CP-7	Alternate Processing Site	Audit virtual machines without disaster recovery configured	1.0.0
Contingency Planning	CP-9	Information System Backup	Azure Backup should be enabled for Virtual Machines	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	IA-5	Authenticator Management	Authentication to Linux machines should require SSH keys	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA-5	Authenticator Management	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have the password complexity setting enabled	2.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Risk Assessment	RA-5	Vulnerability Scanning	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	SQL servers on machines should have vulnerability findings resolved	1.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in container security configurations should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Scanning	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-5	Denial of Service Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7	Boundary Protection	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7	Boundary Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports should be closed on your virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-7 (3)	Access Points	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7 (3)	Access Points	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports should be closed on your virtual machines	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-8	Transmission Confidentiality and Integrity	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC-8 (1)	Cryptographic or Alternate Physical Protection	Windows web servers should be configured to use secure communication protocols	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	Managed disks should be double encrypted with both platform-managed and customer-managed keys	1.0.0
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	OS and data disks should be encrypted with a customer-managed key	3.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Information Integrity	SI-2	Flaw Remediation	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates on virtual machine scale sets should be installed	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates should be installed on your machines	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Information Integrity	SI-3 (1)	Central Management	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3 (1)	Central Management	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3 (1)	Central Management	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Information Integrity	SI-4	Information System Monitoring	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
System and Information Integrity	SI-4	Information System Monitoring	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI-4	Information System Monitoring	Guest Configuration extension should be installed on your machines	1.0.2
System and Information Integrity	SI-4	Information System Monitoring	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
System and Information Integrity	SI-4	Information System Monitoring	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
System and Information Integrity	SI-4	Information System Monitoring	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
System and Information Integrity	SI-16	Memory Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0

## HIPAA HITRUST 9.2

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - HIPAA HITRUST 9.2](#). For more information about this compliance standard, see [HIPAA HITRUST 9.2](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Privilege Management	11180.01c3System.6 - 01.c	Access to management functions or administrative consoles for systems hosting virtualized systems are restricted to personnel based upon the principle of least privilege and supported through technical controls.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Privilege Management	1143.01c1System.123 - 01.c	Privileges are formally authorized and controlled, allocated to users on a need-to-use and event-by-event basis for their functional role (e.g., user or administrator), and documented for each system product/element.	Management ports should be closed on your virtual machines	3.0.0
Privilege Management	1148.01c2System.78 - 01.c	The organization restricts access to privileged functions and all security-relevant information.	Windows machines should meet requirements for 'Security Options - Accounts'	3.0.0
Privilege Management	1150.01c2System.10 - 01.c	The access control system for the system components storing, processing or transmitting covered information is set with a default "deny-all" setting.	Management ports should be closed on your virtual machines	3.0.0
User Authentication for External Connections	1119.01j2Organizational.3 - 01.j	Network equipment is checked for unanticipated dial-up capabilities.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
User Authentication for External Connections	1175.01j1Organizational.8 - 01.j	Remote access to business information across public networks only takes place after successful identification and authentication.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
User Authentication for External Connections	1179.01j3Organizational.1 - 01.j	The information system monitors and controls remote access methods.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Remote Diagnostic and Configuration Port Protection	1192.01l1Organizational.1 - 01.l	Access to network equipment is physically protected.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Remote Diagnostic and Configuration Port Protection	1193.01l2Organizational.13 - 01.l	Controls for the access to diagnostic and configuration ports include the use of a key lock and the implementation of supporting procedures to control physical access to the port.	<a href="#">Management ports should be closed on your virtual machines</a>	3.0.0
Remote Diagnostic and Configuration Port Protection	1197.01l3Organizational.3 - 01.l	The organization disables Bluetooth and peer-to-peer networking protocols within the information system determined to be unnecessary or non-secure.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Segregation in Networks	0805.01m1Organizational.12 - 01.m	The organization's security gateways (e.g. firewalls) enforce security policies and are configured to filter traffic between domains, block unauthorized access, and are used to maintain segregation between internal wired, internal wireless, and external network segments (e.g., the Internet) including DMZs and enforce access control policies for each of the domains.	<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Segregation in Networks	0806.01m2Organizational.12356 - 01.m	The organizations network is logically and physically segmented with a defined security perimeter and a graduated set of controls, including subnetworks for publicly accessible system components that are logically separated from the internal network, based on organizational requirements; and traffic is controlled based on functionality required and classification of the data/systems based on a risk assessment and their respective security requirements.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Segregation in Networks	0894.01m2Organizational.7 - 01.m	Networks are segregated from production-level networks when migrating physical servers, applications or data to virtualized servers.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Network Connection Control	0809.01n2Organizational.1234 - 01.n	Network traffic is controlled in accordance with the organizations access control policy through firewall and other network-related restrictions for each network access point or external telecommunication service's managed interface.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Connection Control	0809.01n2Organizational.1234 - 01.n	Network traffic is controlled in accordance with the organizations access control policy through firewall and other network-related restrictions for each network access point or external telecommunication service's managed interface.	<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	3.0.0
Network Connection Control	0810.01n2Organizational.5 - 01.n	Transmitted information is secured and, at a minimum, encrypted over open, public networks.	<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	3.0.0
Network Connection Control	0810.01n2Organizational.5 - 01.n	Transmitted information is secured and, at a minimum, encrypted over open, public networks.	<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	3.0.0
Network Connection Control	0811.01n2Organizational.6 - 01.n	Exceptions to the traffic flow policy are documented with a supporting mission/business need, duration of the exception, and reviewed at least annually; traffic flow policy exceptions are removed when no longer supported by an explicit mission/business need.	<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Connection Control	0811.01n2Organizational.6 - 01.n	Exceptions to the traffic flow policy are documented with a supporting mission/business need, duration of the exception, and reviewed at least annually; traffic flow policy exceptions are removed when no longer supported by an explicit mission/business need.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Network Connection Control	0812.01n2Organizational.8 - 01.n	Remote devices establishing a non-remote connection are not allowed to communicate with external (remote) resources.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Network Connection Control	0812.01n2Organizational.8 - 01.n	Remote devices establishing a non-remote connection are not allowed to communicate with external (remote) resources.	Internet-facing virtual machines should be protected with network security groups	3.0.0
Network Connection Control	0814.01n1Organizational.12 - 01.n	The ability of users to connect to the internal network is restricted using a deny-by-default and allow-by-exception policy at managed interfaces according to the access control policy and the requirements of clinical and business applications.	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Network Connection Control	0814.01n1Organizational.12 - 01.n	The ability of users to connect to the internal network is restricted using a deny-by-default and allow-by-exception policy at managed interfaces according to the access control policy and the requirements of clinical and business applications.	Internet-facing virtual machines should be protected with network security groups	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
User Identification and Authentication	11210.01q2Organizational.10 - 01.q	Electronic signatures and handwritten signatures executed to electronic records shall be linked to their respective electronic records.	Audit Windows machines that have the specified members in the Administrators group	2.0.0
User Identification and Authentication	11211.01q2Organizational.11 - 01.q	Signed electronic records shall contain information associated with the signing in human-readable format.	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
User Identification and Authentication	1123.01q1System.2 - 01.q	Users who performed privileged functions (e.g., system administration) use separate accounts when performing those privileged functions.	Audit Windows machines that have extra accounts in the Administrators group	2.0.0
User Identification and Authentication	1125.01q2System.1 - 01.q	Multi-factor authentication methods are used in accordance with organizational policy, (e.g., for remote network access).	Audit Windows machines that have the specified members in the Administrators group	2.0.0
User Identification and Authentication	1127.01q2System.3 - 01.q	Where tokens are provided for multi-factor authentication, in-person verification is required prior to granting access.	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Audit Logging	1202.09aa1System.1 - 09.aa	A secure audit record is created for all activities on the system (create, read, update, delete) involving covered information.	System updates on virtual machine scale sets should be installed	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Audit Logging	1206.09aa2System.2 3 - 09.aa	Auditing is always available while the system is active and tracks key events, success/failed data access, system security configuration changes, privileged or utility use, any alarms raised, activation and deactivation of protection systems (e.g., A/V and IDS), activation and deactivation of identification and authentication mechanisms, and creation and deletion of system-level objects.	<a href="#">Resource logs in Virtual Machine Scale Sets should be enabled</a>	2.1.0
Monitoring System Use	12100.09ab2System. 15 - 09.ab	The organization monitors the information system to identify irregularities or anomalies that are indicators of a system malfunction or compromise and help confirm the system is functioning in an optimal, resilient and secure state.	<a href="#">Virtual machines should have the Log Analytics extension installed</a>	1.0.1
Monitoring System Use	12101.09ab1Organizational. 3 - 09.ab	The organization specifies how often audit logs are reviewed, how the reviews are documented, and the specific roles and responsibilities of the personnel conducting the reviews, including the professional certifications or other qualifications required.	<a href="#">The Log Analytics extension should be installed on Virtual Machine Scale Sets</a>	1.0.1

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Monitoring System Use	12102.09ab1Organizational.4 - 09.ab	The organization shall periodically test its monitoring and detection processes, remediate deficiencies, and improve its processes.	<a href="#">Audit Windows machines on which the Log Analytics agent is not connected as expected</a>	2.0.0
Monitoring System Use	1215.09ab2System.7 - 09.ab	Auditing and monitoring systems employed by the organization support audit reduction and report generation.	<a href="#">Virtual machines should have the Log Analytics extension installed</a>	1.0.1
Monitoring System Use	1216.09ab3System.12 - 09.ab	Automated systems are used to review monitoring activities of security systems (e.g., IPS/IDS) and system records on a daily basis, and identify and document anomalies.	<a href="#">The Log Analytics extension should be installed on Virtual Machine Scale Sets</a>	1.0.1
Monitoring System Use	1217.09ab3System.3 - 09.ab	Alerts are generated for technical personnel to analyze and investigate suspicious activity or suspected violations.	<a href="#">Audit Windows machines on which the Log Analytics agent is not connected as expected</a>	2.0.0
Segregation of Duties	1232.09c3Organizational.12 - 09.c	Access for individuals responsible for administering access controls is limited to the minimum necessary based upon each user's role and responsibilities and these individuals cannot access audit functions related to these controls.	<a href="#">Windows machines should meet requirements for 'User Rights Assignment'</a>	3.0.0
Segregation of Duties	1277.09c2Organizational.4 - 09.c	The initiation of an event is separated from its authorization to reduce the possibility of collusion.	<a href="#">Windows machines should meet requirements for 'Security Options - User Account Control'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">Deploy default Microsoft IaaSAntimalware extension for Windows Server</a>	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">Endpoint protection solution should be installed on virtual machine scale sets</a>	3.0.0
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">Microsoft Antimalware for Azure should be configured to automatically update protection signatures</a>	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
Controls Against Malicious Code	0201.09j1Organizational.124 - 09.j	Anti-virus and anti-spyware are installed, operating and updated on all end-user devices to conduct periodic scans of the systems to identify and remove unauthorized software. Server environments for which the server software developer specifically recommends not installing host-based anti-virus and anti-spyware software may address the requirement via a network-based malware detection (NBMD) solution.	<a href="#">System updates should be installed on your machines</a>	4.0.0
Back-up	1620.09l1Organizational.8 - 09.l	When the backup service is delivered by the third party, the service level agreement includes the detailed protections to control confidentiality, integrity and availability of the backup information.	<a href="#">Azure Backup should be enabled for Virtual Machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Back-up	1625.09l3Organizational.34 - 09.l	Three (3) generations of backups (full plus all related incremental or differential backups) are stored off-site, and both on-site and off-site backups are logged with name, date, time and action.	Azure Backup should be enabled for Virtual Machines	3.0.0
Back-up	1699.09l1Organizational.10 - 09.l	Workforce members roles and responsibilities in the data backup process are identified and communicated to the workforce; in particular, Bring Your Own Device (BYOD) users are required to perform backups of organizational and/or client data on their devices.	Azure Backup should be enabled for Virtual Machines	3.0.0
Network Controls	0858.09m1Organizational.4 - 09.m	The organization monitors for all authorized and unauthorized wireless access to the information system and prohibits installation of wireless access points (WAPs) unless explicitly authorized in writing by the CIO or his/her designated representative.	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Network Controls	0858.09m1Organizational.4 - 09.m	The organization monitors for all authorized and unauthorized wireless access to the information system and prohibits installation of wireless access points (WAPs) unless explicitly authorized in writing by the CIO or his/her designated representative.	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Controls	0858.09m1Organizational.4 - 09.m	The organization monitors for all authorized and unauthorized wireless access to the information system and prohibits installation of wireless access points (WAPs) unless explicitly authorized in writing by the CIO or his/her designated representative.	<a href="#">Windows machines should meet requirements for 'Windows Firewall Properties'</a>	3.0.0
Network Controls	0859.09m1Organizational.78 - 09.m	The organization ensures the security of information in networks, availability of network services and information services using the network, and the protection of connected services from unauthorized access.	<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	3.0.0
Network Controls	0861.09m2Organizational.67 - 09.m	To identify and authenticate devices on local and/or wide area networks, including wireless networks, the information system uses either a (i) shared known information solution or (ii) an organizational authentication solution, the exact selection and strength of which is dependent on the security categorization of the information system.	<a href="#">Windows machines should meet requirements for 'Security Options - Network Access'</a>	3.0.0
Security of Network Services	0835.09n1Organizational.1 - 09.n	Agreed services provided by a network service provider/manager are formally managed and monitored to ensure they are provided securely.	<a href="#">[Preview]: Network traffic data collection agent should be installed on Windows virtual machines</a>	1.0.2-preview

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Security of Network Services	0835.09n1Organizational.1 - 09.n	Agreed services provided by a network service provider/manager are formally managed and monitored to ensure they are provided securely.	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Security of Network Services	0836.09.n2Organizational.1 - 09.n	The organization formally authorizes and documents the characteristics of each connection from an information system to other information systems outside the organization.	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Security of Network Services	0885.09n2Organizational.3 - 09.n	The organization reviews and updates the interconnection security agreements on an ongoing basis verifying enforcement of security requirements.	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Security of Network Services	0887.09n2Organizational.5 - 09.n	The organization requires external/outsourced service providers to identify the specific functions, ports, and protocols used in the provision of the external/outsourced services.	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Management of Removable Media	0302.09o2Organizational.1 - 09.o	The organization protects and controls media containing sensitive information during transport outside of controlled areas.	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
On-line Transactions	0945.09y1Organizational.3 - 09.y	Protocols used to communicate between all involved parties are secured using cryptographic techniques (e.g., SSL).	Audit Windows machines that do not contain the specified certificates in Trusted Root	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Control of Operational Software	0605.10h1System.12 - 10.h	Only authorized administrators are allowed to implement approved upgrades to software, applications, and program libraries, based on business requirements and the security implications of the release.	<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	3.0.0
Control of Operational Software	0605.10h1System.12 - 10.h	Only authorized administrators are allowed to implement approved upgrades to software, applications, and program libraries, based on business requirements and the security implications of the release.	<a href="#">Windows machines should meet requirements for 'Security Options - Audit'</a>	3.0.0
Control of Operational Software	0605.10h1System.12 - 10.h	Only authorized administrators are allowed to implement approved upgrades to software, applications, and program libraries, based on business requirements and the security implications of the release.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Account Management'</a>	3.0.0
Control of Operational Software	0606.10h2System.1 - 10.h	Applications and operating systems are successfully tested for usability, security and impact prior to production.	<a href="#">Vulnerabilities in container security configurations should be remediated</a>	3.0.0
Control of Operational Software	0607.10h2System.23 - 10.h	The organization uses its configuration control program to maintain control of all implemented software and its system documentation and archive prior versions of implemented software and associated system documentation.	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Control of Operational Software	0607.10h2System.23 - 10.h	The organization uses its configuration control program to maintain control of all implemented software and its system documentation and archive prior versions of implemented software and associated system documentation.	<a href="#">Vulnerabilities in security configuration on your virtual machine scale sets should be remediated</a>	3.0.0
Change Control Procedures	0635.10k1Organizational.12 - 10.k	Managers responsible for application systems are also responsible for the strict control (security) of the project or support environment and ensure that all proposed system changes are reviewed to check that they do not compromise the security of either the system or the operating environment.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0636.10k2Organizational.1 - 10.k	The organization formally addresses purpose, scope, roles, responsibilities, management commitment, coordination among organizational entities, and compliance for configuration management (e.g., through policies, standards, processes).	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0637.10k2Organizational.2 - 10.k	The organization has developed, documented, and implemented a configuration management plan for the information system.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Change Control Procedures	0638.10k2Organizational.34569 - 10.k	Changes are formally controlled, documented and enforced in order to minimize the corruption of information systems.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0639.10k2Organizational.78 - 10.k	Installation checklists and vulnerability scans are used to validate the configuration of servers, workstations, devices and appliances and ensure the configuration meets minimum standards.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0640.10k2Organizational.1012 - 10.k	Where development is outsourced, change control procedures to address security are included in the contract(s) and specifically require the developer to track security flaws and flaw resolution within the system, component, or service and report findings to organization-defined personnel or roles.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0641.10k2Organizational.11 - 10.k	The organization does not use automated updates on critical systems.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0642.10k3Organizational.12 - 10.k	The organization develops, documents, and maintains, under configuration control, a current baseline configuration of the information system, and reviews and updates the baseline as required.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Change Control Procedures	0643.10k3Organizational.3 - 10.k	The organization (i) establishes and documents mandatory configuration settings for information technology products employed within the information system using the latest security configuration baselines; (ii) identifies, documents, and approves exceptions from the mandatory established configuration settings for individual components based on explicit operational requirements; and (iii) monitors and controls changes to the configuration settings in accordance with organizational policies and procedures.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Change Control Procedures	0644.10k3Organizational.4 - 10.k	The organization employs automated mechanisms to (i) centrally manage, apply, and verify configuration settings; (ii) respond to unauthorized changes to network and system security-related configuration settings; and (iii) enforce access restrictions and auditing of the enforcement actions.	<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	3.0.0
Control of Technical Vulnerabilities	0709.10m1Organizational.1 - 10.m	Technical vulnerabilities are identified, evaluated for risk and corrected in a timely manner.	<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Control of Technical Vulnerabilities	0709.10m1Organizational.1 - 10.m	Technical vulnerabilities are identified, evaluated for risk and corrected in a timely manner.	Vulnerabilities in container security configurations should be remediated	3.0.0
Control of Technical Vulnerabilities	0709.10m1Organizational.1 - 10.m	Technical vulnerabilities are identified, evaluated for risk and corrected in a timely manner.	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Control of Technical Vulnerabilities	0709.10m1Organizational.1 - 10.m	Technical vulnerabilities are identified, evaluated for risk and corrected in a timely manner.	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Control of Technical Vulnerabilities	0709.10m1Organizational.1 - 10.m	Technical vulnerabilities are identified, evaluated for risk and corrected in a timely manner.	Windows machines should meet requirements for 'Security Options - Microsoft Network Server'	3.0.0
Control of Technical Vulnerabilities	0711.10m2Organizational.23 - 10.m	A technical vulnerability management program is in place to monitor, assess, rank, and remediate vulnerabilities identified in systems.	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Control of Technical Vulnerabilities	0713.10m2Organizational.5 - 10.m	Patches are tested and evaluated before they are installed.	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Control of Technical Vulnerabilities	0714.10m2Organizational.7 - 10.m	The technical vulnerability management program is evaluated on a quarterly basis.	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Control of Technical Vulnerabilities	0715.10m2Organizational.8 - 10.m	Systems are appropriately hardened (e.g., configured with only necessary and secure services, ports and protocols enabled).	Vulnerabilities in container security configurations should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Control of Technical Vulnerabilities	0717.10m3Organizational.2 - 10.m	Vulnerability scanning tools include the capability to readily update the information system vulnerabilities scanned.	<a href="#">Vulnerabilities in security configuration on your virtual machine scale sets should be remediated</a>	3.0.0
Control of Technical Vulnerabilities	0718.10m3Organizational.34 - 10.m	The organization scans for vulnerabilities in the information system and hosted applications to determine the state of flaw remediation monthly (automatically) and again (manually or automatically) when new vulnerabilities potentially affecting the systems and networked environments are identified and reported.	<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	3.0.0
Business Continuity and Risk Assessment	1634.12b1Organizational.1 - 12.b	The organization identifies the critical business processes requiring business continuity.	<a href="#">Audit virtual machines without disaster recovery configured</a>	1.0.0
Business Continuity and Risk Assessment	1637.12b2Organizational.2 - 12.b	Business impact analysis are used to evaluate the consequences of disasters, security failures, loss of service, and service availability.	<a href="#">Windows machines should meet requirements for 'Security Options - Recovery console'</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Business Continuity and Risk Assessment	1638.12b2Organizational.345 - 12.b	Business continuity risk assessments (i) are carried out annually with full involvement from owners of business resources and processes; (ii) consider all business processes and is not limited to the information assets, but includes the results specific to information security; and (iii) identifies, quantifies, and prioritizes risks against key business objectives and criteria relevant to the organization, including critical resources, impacts of disruptions, allowable outage times, and recovery priorities.	<a href="#">Audit virtual machines without disaster recovery configured</a>	1.0.0

## IRS 1075 September 2016

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - IRS 1075 September 2016](#). For more information about this compliance standard, see [IRS 1075 September 2016](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	9.3.1.12	Remote Access (AC-17)	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access Control	9.3.1.12	Remote Access (AC-17)	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	9.3.1.12	Remote Access (AC-17)	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	9.3.1.12	Remote Access (AC-17)	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	9.3.1.2	Account Management (AC-2)	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	9.3.1.5	Separation of Duties (AC-5)	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	9.3.1.5	Separation of Duties (AC-5)	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	9.3.1.5	Separation of Duties (AC-5)	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Access Control	9.3.1.5	Separation of Duties (AC-5)	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Access Control	9.3.1.5	Separation of Duties (AC-5)	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	9.3.1.6	Least Privilege (AC-6)	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	9.3.1.6	Least Privilege (AC-6)	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	9.3.1.6	Least Privilege (AC-6)	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Access Control	9.3.1.6	Least Privilege (AC-6)	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Access Control	9.3.1.6	Least Privilege (AC-6)	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Risk Assessment	9.3.14.3	Vulnerability Scanning (RA-5)	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	9.3.14.3	Vulnerability Scanning (RA-5)	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	9.3.14.3	Vulnerability Scanning (RA-5)	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	9.3.16.15	Protection of Information at Rest (SC-28)	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Communications Protection	9.3.16.5	Boundary Protection (SC-7)	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
System and Communications Protection	9.3.16.5	Boundary Protection (SC-7)	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	9.3.16.5	Boundary Protection (SC-7)	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	9.3.16.6	Transmission Confidentiality and Integrity (SC-8)	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
System and Communications Protection	9.3.16.6	Transmission Confidentiality and Integrity (SC-8)	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
System and Communications Protection	9.3.16.6	Transmission Confidentiality and Integrity (SC-8)	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
System and Communications Protection	9.3.16.6	Transmission Confidentiality and Integrity (SC-8)	Windows web servers should be configured to use secure communication protocols	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	9.3.17.2	Flaw Remediation (SI-2)	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
System and Information Integrity	9.3.17.2	Flaw Remediation (SI-2)	System updates on virtual machine scale sets should be installed	3.0.0
System and Information Integrity	9.3.17.2	Flaw Remediation (SI-2)	System updates should be installed on your machines	4.0.0
System and Information Integrity	9.3.17.2	Flaw Remediation (SI-2)	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
System and Information Integrity	9.3.17.2	Flaw Remediation (SI-2)	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Information Integrity	9.3.17.3	Malicious Code Protection (SI-3)	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	9.3.17.3	Malicious Code Protection (SI-3)	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	9.3.17.4	Information System Monitoring (SI-4)	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
System and Information Integrity	9.3.17.4	Information System Monitoring (SI-4)	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
System and Information Integrity	9.3.17.4	Information System Monitoring (SI-4)	Virtual machines should be connected to a specified workspace	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Awareness and Training	9.3.3.11	Audit Generation (AU-12)	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Awareness and Training	9.3.3.11	Audit Generation (AU-12)	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Awareness and Training	9.3.3.11	Audit Generation (AU-12)	Virtual machines should be connected to a specified workspace	1.1.0
Awareness and Training	9.3.3.3	Content of Audit Records (AU-3)	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Awareness and Training	9.3.3.3	Content of Audit Records (AU-3)	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Awareness and Training	9.3.3.3	Content of Audit Records (AU-3)	Virtual machines should be connected to a specified workspace	1.1.0
Awareness and Training	9.3.3.6	Audit Review, Analysis, and Reporting (AU-6)	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Awareness and Training	9.3.3.6	Audit Review, Analysis, and Reporting (AU-6)	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Awareness and Training	9.3.3.6	Audit Review, Analysis, and Reporting (AU-6)	Virtual machines should be connected to a specified workspace	1.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Configuration Management	9.3.5.11	User-Installed Software (CM-11)	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	9.3.5.7	Least Functionality (CM-7)	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Contingency Planning	9.3.6.6	Alternate Processing Site (CP-7)	Audit virtual machines without disaster recovery configured	1.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Linux machines that have accounts without passwords	3.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that do not have the password complexity setting enabled	2.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	9.3.7.5	Authenticator Management (IA-5)	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

## ISO 27001:2013

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - ISO 27001:2013](#). For more information about this compliance standard, see [ISO 27001:2013](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Cryptography	10.1.1	Policy on the use of cryptographic controls	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Cryptography	10.1.1	Policy on the use of cryptographic controls	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Cryptography	10.1.1	Policy on the use of cryptographic controls	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Cryptography	10.1.1	Policy on the use of cryptographic controls	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Cryptography	10.1.1	Policy on the use of cryptographic controls	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Operations security	12.4.1	Event Logging	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Operations security	12.4.1	Event Logging	Dependency agent should be enabled for listed virtual machine images	2.0.0
Operations security	12.4.1	Event Logging	Dependency agent should be enabled in virtual machine scale sets for listed virtual machine images	2.0.0
Operations security	12.4.1	Event Logging	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Operations security	12.4.3	Administrator and operator logs	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Operations security	12.4.3	Administrator and operator logs	Dependency agent should be enabled for listed virtual machine images	2.0.0
Operations security	12.4.3	Administrator and operator logs	Dependency agent should be enabled in virtual machine scale sets for listed virtual machine images	2.0.0
Operations security	12.4.3	Administrator and operator logs	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Operations security	12.4.4	Clock Synchronization	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Operations security	12.4.4	Clock Synchronization	Dependency agent should be enabled for listed virtual machine images	2.0.0
Operations security	12.4.4	Clock Synchronization	Dependency agent should be enabled in virtual machine scale sets for listed virtual machine images	2.0.0
Operations security	12.4.4	Clock Synchronization	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Operations security	12.5.1	Installation of software on operational systems	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Operations security	12.6.1	Management of technical vulnerabilities	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Operations security	12.6.1	Management of technical vulnerabilities	<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	3.0.0
Operations security	12.6.1	Management of technical vulnerabilities	<a href="#">System updates should be installed on your machines</a>	4.0.0
Operations security	12.6.1	Management of technical vulnerabilities	<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	3.0.0
Operations security	12.6.2	Restrictions on software installation	<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	3.0.0
Communications security	13.1.1	Network controls	<a href="#">All network ports should be restricted on network security groups associated to your virtual machine</a>	3.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Audit Linux machines that allow remote connections from accounts without passwords</a>	3.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Audit Linux machines that have accounts without passwords</a>	3.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Audit VMs that do not use managed disks</a>	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access control	9.1.2	Access to networks and network services	<a href="#">Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs</a>	3.0.0
Access control	9.1.2	Access to networks and network services	<a href="#">Virtual machines should be migrated to new Azure Resource Manager resources</a>	1.0.0
Access control	9.2.4	Management of secret authentication information of users	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access control	9.2.4	Management of secret authentication information of users	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0
Access control	9.2.4	Management of secret authentication information of users	<a href="#">Audit Linux machines that do not have the passwd file permissions set to 0644</a>	3.0.0
Access control	9.2.4	Management of secret authentication information of users	<a href="#">Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs</a>	3.0.0
Access control	9.4.3	Password management system	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	4.0.0
Access control	9.4.3	Password management system	<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access control	9.4.3	Password management system	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Access control	9.4.3	Password management system	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Access control	9.4.3	Password management system	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Access control	9.4.3	Password management system	Audit Windows machines that do not have the password complexity setting enabled	2.0.0
Access control	9.4.3	Password management system	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Access control	9.4.3	Password management system	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

## New Zealand ISM Restricted

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - New Zealand ISM Restricted](#). For more information about this compliance standard, see [New Zealand ISM Restricted](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Information security monitoring	ISM-3	6.2.5 Conducting vulnerability assessments	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Information security monitoring	ISM-4	6.2.6 Resolving vulnerabilities	SQL servers on machines should have vulnerability findings resolved	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Information security monitoring	ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in container security configurations should be remediated	3.0.0
Information security monitoring	ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Information security monitoring	ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Information security monitoring	ISM-7	6.4.5 Availability requirements	Audit virtual machines without disaster recovery configured	1.0.0
Product Security	PRS-5	12.4.4 Patching vulnerabilities in products	System updates on virtual machine scale sets should be installed	3.0.0
Product Security	PRS-5	12.4.4 Patching vulnerabilities in products	System updates should be installed on your machines	4.0.0
Software security	SS-2	14.1.8 Developing hardened SOEs	Management ports should be closed on your virtual machines	3.0.0
Software security	SS-3	14.1.9 Maintaining hardened SOEs	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Software security	SS-3	14.1.9 Maintaining hardened SOEs	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Software security	SS-3	14.1.9 Maintaining hardened SOEs	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Software security	SS-3	14.1.9 Maintaining hardened SOEs	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Software security	SS-3	14.1.9 Maintaining hardened SOEs	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Software security	SS-5	14.2.4 Application Whitelisting	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Software security	SS-5	14.2.4 Application Whitelisting	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Access Control and Passwords	AC-4	16.1.40 Password selection policy	Audit Linux machines that have accounts without passwords	3.0.0
Access Control and Passwords	AC-4	16.1.40 Password selection policy	Windows machines should meet requirements for 'Security Settings - Account Policies'	3.0.0
Access Control and Passwords	AC-11	16.4.30 Privileged Access Management	Audit Windows machines missing any of specified members in the Administrators group	2.0.0
Access Control and Passwords	AC-11	16.4.30 Privileged Access Management	Audit Windows machines that have extra accounts in the Administrators group	2.0.0
Access Control and Passwords	AC-11	16.4.30 Privileged Access Management	Audit Windows machines that have the specified members in the Administrators group	2.0.0
Access Control and Passwords	AC-13	16.5.10 Authentication	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control and Passwords	AC-17	16.6.9 Events to be logged	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Access Control and Passwords	AC-17	16.6.9 Events to be logged	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Cryptography	CR-3	17.1.46 Reducing storage and physical transfer requirements	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Cryptography	CR-7	17.4.16 Using TLS	Windows web servers should be configured to use secure communication protocols	4.0.0
Cryptography	CR-9	17.5.7 Authentication mechanisms	Authentication to Linux machines should require SSH keys	3.0.0
Cryptography	CR-14	17.9.25 Contents of KMPs	IP Forwarding on your virtual machine should be disabled	3.0.0
Gateway security	GS-2	19.1.11 Using Gateways	Internet-facing virtual machines should be protected with network security groups	3.0.0
Gateway security	GS-3	19.1.12 Configuration of Gateways	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Gateway security	GS-5	19.1.23 Testing of Gateways	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NIST SP 800-53 Rev. 5](#). For more information about this compliance standard, see [NIST SP 800-53 Rev. 5](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control	AC-2 (12)	Account Monitoring for Atypical Usage	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC-3	Access Enforcement	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-3	Access Enforcement	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-3	Access Enforcement	Audit Linux machines that have accounts without passwords	3.0.0
Access Control	AC-3	Access Enforcement	Authentication to Linux machines should require SSH keys	3.0.0
Access Control	AC-3	Access Enforcement	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-3	Access Enforcement	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Access Control	AC-4	Information Flow Enforcement	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC-4	Information Flow Enforcement	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Access Control	AC-4	Information Flow Enforcement	Disk access resources should use private link	1.0.0
Access Control	AC-4	Information Flow Enforcement	Internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-4	Information Flow Enforcement	IP Forwarding on your virtual machine should be disabled	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	AC-4	Information Flow Enforcement	Management ports should be closed on your virtual machines	3.0.0
Access Control	AC-4	Information Flow Enforcement	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Access Control	AC-4 (3)	Dynamic Information Flow Control	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Access Control	AC-4 (3)	Dynamic Information Flow Control	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17	Remote Access	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Access Control	AC-17	Remote Access	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17	Remote Access	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-17	Remote Access	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17	Remote Access	Disk access resources should use private link	1.0.0
Access Control	AC-17 (1)	Monitoring and Control	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Access Control	AC-17 (1)	Monitoring and Control	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Access Control	AC-17 (1)	Monitoring and Control	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control	AC-17 (1)	Monitoring and Control	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Access Control	AC-17 (1)	Monitoring and Control	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Access Control	AC-17 (1)	Monitoring and Control	Disk access resources should use private link	1.0.0
Audit and Accountability	AU-6	Audit Record Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6	Audit Record Review, Analysis, and Reporting	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (4)	Central Review and Analysis	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (4)	Central Review and Analysis	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Guest Configuration extension should be installed on your machines	1.0.2

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-6 (4)	Central Review and Analysis	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-6 (5)	Integrated Analysis of Audit Records	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Audit and Accountability	AU-12	Audit Record Generation	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Record Generation	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12	Audit Record Generation	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-12	Audit Record Generation	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Record Generation	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12	Audit Record Generation	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-12	Audit Record Generation	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1

Domain	Control ID	Control Title	Policy	Policy Version
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	Guest Configuration extension should be installed on your machines	1.0.2
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Audit and Accountability	AU-12 (1)	System-wide and Time-correlated Audit Trail	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Configuration Management	CM-6	Configuration Settings	Linux machines should meet requirements for the Azure compute security baseline	2.0.0
Configuration Management	CM-6	Configuration Settings	Windows machines should meet requirements of the Azure compute security baseline	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Configuration Management	CM-7	Least Functionality	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7	Least Functionality	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (2)	Prevent Program Execution	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-7 (5)	Authorized Software ??? Allow-by-exception	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-7 (5)	Authorized Software ??? Allow-by-exception	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-10	Software Usage Restrictions	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Configuration Management	CM-11	User-installed Software	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Configuration Management	CM-11	User-installed Software	Allowlist rules in your adaptive application control policy should be updated	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Contingency Planning	CP-7	Alternate Processing Site	Audit virtual machines without disaster recovery configured	1.0.0
Contingency Planning	CP-9	System Backup	Azure Backup should be enabled for Virtual Machines	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Audit Windows machines that do not store passwords using reversible encryption	2.0.0
Identification and Authentication	IA-5	Authenticator Management	Authentication to Linux machines should require SSH keys	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	IA-5	Authenticator Management	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not have the password complexity setting enabled	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Audit Windows machines that do not store passwords using reversible encryption	2.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identification and Authentication	IA-5 (1)	Password-based Authentication	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Risk Assessment	RA-5	Vulnerability Monitoring and Scanning	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Risk Assessment	RA-5	Vulnerability Monitoring and Scanning	SQL servers on machines should have vulnerability findings resolved	1.0.0
Risk Assessment	RA-5	Vulnerability Monitoring and Scanning	Vulnerabilities in container security configurations should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Monitoring and Scanning	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Risk Assessment	RA-5	Vulnerability Monitoring and Scanning	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Communications Protection	SC-3	Security Function Isolation	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Communications Protection	SC-3	Security Function Isolation	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Communications Protection	SC-3	Security Function Isolation	Windows Defender Exploit Guard should be enabled on your machines	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-5	Denial-of-service Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7	Boundary Protection	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7	Boundary Protection	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Management ports should be closed on your virtual machines	3.0.0
System and Communications Protection	SC-7	Boundary Protection	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Communications Protection	SC-7 (3)	Access Points	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Disk access resources should use private link	1.0.0
System and Communications Protection	SC-7 (3)	Access Points	Internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	IP Forwarding on your virtual machine should be disabled	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Management ports should be closed on your virtual machines	3.0.0
System and Communications Protection	SC-7 (3)	Access Points	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
System and Communications Protection	SC-8	Transmission Confidentiality and Integrity	Windows web servers should be configured to use secure communication protocols	4.0.0
System and Communications Protection	SC-8 (1)	Cryptographic Protection	Windows web servers should be configured to use secure communication protocols	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	Managed disks should be double encrypted with both platform-managed and customer-managed keys	1.0.0
System and Communications Protection	SC-12	Cryptographic Key Establishment and Management	OS and data disks should be encrypted with a customer-managed key	3.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28	Protection of Information at Rest	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
System and Communications Protection	SC-28 (1)	Cryptographic Protection	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
System and Information Integrity	SI-2	Flaw Remediation	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates on virtual machine scale sets should be installed	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	System updates should be installed on your machines	4.0.0

Domain	Control ID	Control Title	Policy	Policy Version
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
System and Information Integrity	SI-2	Flaw Remediation	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
System and Information Integrity	SI-3	Malicious Code Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
System and Information Integrity	SI-4	System Monitoring	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
System and Information Integrity	SI-4	System Monitoring	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
System and Information Integrity	SI-4	System Monitoring	Guest Configuration extension should be installed on your machines	1.0.2
System and Information Integrity	SI-4	System Monitoring	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
System and Information Integrity	SI-4	System Monitoring	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
System and Information Integrity	SI-4	System Monitoring	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
System and Information Integrity	SI-16	Memory Protection	Windows Defender Exploit Guard should be enabled on your machines	2.0.0

## NZ ISM Restricted v3.5

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - NZ ISM Restricted v3.5](#). For more information about this compliance standard, see [NZ ISM Restricted v3.5](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Access Control and Passwords	NZISM Security Benchmark AC-13	16.5.10 Authentication	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Access Control and Passwords	NZISM Security Benchmark AC-18	16.6.9 Events to be logged	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Access Control and Passwords	NZISM Security Benchmark AC-18	16.6.9 Events to be logged	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Access Control and Passwords	NZISM Security Benchmark AC-18	16.6.9 Events to be logged	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Cryptography	NZISM Security Benchmark CR-10	17.5.7 Authentication mechanisms	<a href="#">Authentication to Linux machines should require SSH keys</a>	3.0.0
Cryptography	NZISM Security Benchmark CR-15	17.9.25 Contents of KMPs	<a href="#">IP Forwarding on your virtual machine should be disabled</a>	3.0.0
Cryptography	NZISM Security Benchmark CR-3	17.1.53 Reducing storage and physical transfer requirements	<a href="#">Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources</a>	2.0.3
Cryptography	NZISM Security Benchmark CR-8	17.4.16 Using TLS	<a href="#">Windows web servers should be configured to use secure communication protocols</a>	4.0.0
Gateway security	NZISM Security Benchmark GS-2	19.1.11 Using Gateways	<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	3.0.0
Gateway security	NZISM Security Benchmark GS-2	19.1.11 Using Gateways	<a href="#">Non-internet-facing virtual machines should be protected with network security groups</a>	3.0.0
Gateway security	NZISM Security Benchmark GS-3	19.1.12 Configuration of Gateways	<a href="#">All network ports should be restricted on network security groups associated to your virtual machine</a>	3.0.0
Gateway security	NZISM Security Benchmark GS-5	19.1.23 Testing of Gateways	<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	3.0.0
Information security monitoring	NZISM Security Benchmark ISM-3	6.2.5 Conducting vulnerability assessments	<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Information security monitoring	NZISM Security Benchmark ISM-4	6.2.6 Resolving vulnerabilities	SQL servers on machines should have vulnerability findings resolved	1.0.0
Information security monitoring	NZISM Security Benchmark ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in container security configurations should be remediated	3.0.0
Information security monitoring	NZISM Security Benchmark ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Information security monitoring	NZISM Security Benchmark ISM-4	6.2.6 Resolving vulnerabilities	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Information security monitoring	NZISM Security Benchmark ISM-7	6.4.5 Availability requirements	Audit virtual machines without disaster recovery configured	1.0.0
Product Security	NZISM Security Benchmark PRS-5	12.4.4 Patching vulnerabilities in products	System updates on virtual machine scale sets should be installed	3.0.0
Product Security	NZISM Security Benchmark PRS-5	12.4.4 Patching vulnerabilities in products	System updates should be installed on your machines	4.0.0
Software security	NZISM Security Benchmark SS-2	14.1.8 Developing hardened SOEs	Management ports should be closed on your virtual machines	3.0.0
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Endpoint protection health issues should be resolved on your machines	1.0.0
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Endpoint protection should be installed on your machines	1.0.0
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Guest Configuration extension should be installed on your machines	1.0.2
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Software security	NZISM Security Benchmark SS-3	14.1.9 Maintaining hardened SOEs	Windows Defender Exploit Guard should be enabled on your machines	2.0.0
Software security	NZISM Security Benchmark SS-5	14.2.4 Application Whitelisting	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Software security	NZISM Security Benchmark SS-5	14.2.4 Application Whitelisting	Allowlist rules in your adaptive application control policy should be updated	3.0.0

## PCI DSS 3.2.1

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [PCI DSS 3.2.1](#). For more information about this compliance standard, see [PCI DSS 3.2.1](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Requirement 1	PCI DSS v3.2.1 1.3.2	PCI DSS requirement 1.3.2	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Requirement 1	PCI DSS v3.2.1 1.3.4	PCI DSS requirement 1.3.4	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Requirement 1	PCI DSS v3.2.1 1.3.4	PCI DSS requirement 1.3.4	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Requirement 10	PCI DSS v3.2.1 10.5.4	PCI DSS requirement 10.5.4	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Requirement 11	PCI DSS v3.2.1 11.2.1	PCI DSS requirement 11.2.1	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Requirement 11	PCI DSS v3.2.1 11.2.1	PCI DSS requirement 11.2.1	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Requirement 11	PCI DSS v3.2.1 11.2.1	PCI DSS requirement 11.2.1	System updates should be installed on your machines	4.0.0
Requirement 11	PCI DSS v3.2.1 11.2.1	PCI DSS requirement 11.2.1	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Requirement 3	PCI DSS v3.2.1 3.4	PCI DSS requirement 3.4	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Requirement 4	PCI DSS v3.2.1 4.1	PCI DSS requirement 4.1	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Requirement 5	PCI DSS v3.2.1 5.1	PCI DSS requirement 5.1	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Requirement 5	PCI DSS v3.2.1 5.1	PCI DSS requirement 5.1	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Requirement 5	PCI DSS v3.2.1 5.1	PCI DSS requirement 5.1	System updates should be installed on your machines	4.0.0
Requirement 5	PCI DSS v3.2.1 5.1	PCI DSS requirement 5.1	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Requirement 6	PCI DSS v3.2.1 6.2	PCI DSS requirement 6.2	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Requirement 6	PCI DSS v3.2.1 6.2	PCI DSS requirement 6.2	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Requirement 6	PCI DSS v3.2.1 6.2	PCI DSS requirement 6.2	System updates should be installed on your machines	4.0.0
Requirement 6	PCI DSS v3.2.1 6.2	PCI DSS requirement 6.2	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Requirement 6	PCI DSS v3.2.1 6.5.3	PCI DSS requirement 6.5.3	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Requirement 6	PCI DSS v3.2.1 6.6	PCI DSS requirement 6.6	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Requirement 6	PCI DSS v3.2.1 6.6	PCI DSS requirement 6.6	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Requirement 6	PCI DSS v3.2.1 6.6	PCI DSS requirement 6.6	System updates should be installed on your machines	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Requirement 6	PCI DSS v3.2.1 6.6	PCI DSS requirement 6.6	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.3	PCI DSS requirement 8.2.3	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Requirement 8	PCI DSS v3.2.1 8.2.5	PCI DSS requirement 8.2.5	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0

## Reserve Bank of India - IT Framework for NBFC

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - Reserve Bank of India - IT Framework for NBFC](#). For more information about this compliance standard, see [Reserve Bank of India - IT Framework for NBFC](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
IT Governance	RBI IT Framework 1	IT Governance-1	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
IT Governance	RBI IT Framework 1	IT Governance-1	SQL servers on machines should have vulnerability findings resolved	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
IT Governance	RBI IT Framework 1	IT Governance-1	System updates on virtual machine scale sets should be installed	3.0.0
IT Governance	RBI IT Framework 1	IT Governance-1	System updates should be installed on your machines	4.0.0
IT Governance	RBI IT Framework 1	IT Governance-1	Vulnerabilities in container security configurations should be remediated	3.0.0
IT Governance	RBI IT Framework 1	IT Governance-1	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
IT Governance	RBI IT Framework 1	IT Governance-1	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
IT Governance	RBI IT Framework 1.1	IT Governance-1.1	IP Forwarding on your virtual machine should be disabled	3.0.0
IT Governance	RBI IT Framework 1.1	IT Governance-1.1	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
IT Governance	RBI IT Framework 1.1	IT Governance-1.1	Management ports should be closed on your virtual machines	3.0.0
IT Policy	RBI IT Framework 2	IT Policy-2	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
IT Policy	RBI IT Framework 2	IT Policy-2	Allowlist rules in your adaptive application control policy should be updated	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Information and Cyber Security	RBI IT Framework 3.1.b	Segregation of Functions-3.1	[Preview]: Secure Boot should be enabled on supported Windows virtual machines	3.0.0-preview
Information and Cyber Security	RBI IT Framework 3.1.b	Segregation of Functions-3.1	[Preview]: vTPM should be enabled on supported virtual machines	2.0.0-preview
Information and Cyber Security	RBI IT Framework 3.1.b	Segregation of Functions-3.1	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Information and Cyber Security	RBI IT Framework 3.1.c	Role based Access Control-3.1	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	[Preview]: Log Analytics Extension should be enabled for listed virtual machine images	2.0.1-preview
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	[Preview]: Network traffic data collection agent should be installed on Windows virtual machines	1.0.2-preview
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	The Log Analytics extension should be installed on Virtual Machine Scale Sets	1.0.1
Information and Cyber Security	RBI IT Framework 3.1.g	Trails-3.1	Virtual machines should have the Log Analytics extension installed	1.0.1
Information and Cyber Security	RBI IT Framework 3.1.h	Public Key Infrastructure (PKI)-3.1	Managed disks should use a specific set of disk encryption sets for the customer-managed key encryption	2.0.0
Information and Cyber Security	RBI IT Framework 3.1.h	Public Key Infrastructure (PKI)-3.1	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	SQL servers on machines should have vulnerability findings resolved	1.0.0
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	System updates on virtual machine scale sets should be installed	3.0.0
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	System updates should be installed on your machines	4.0.0
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	Vulnerabilities in container security configurations should be remediated	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Information and Cyber Security	RBI IT Framework 3.3	Vulnerability Management-3.3	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
IT Operations	RBI IT Framework 4.2	IT Operations-4.2	[Preview]: Network traffic data collection agent should be installed on Linux virtual machines	1.0.2-preview
IT Operations	RBI IT Framework 4.4.a	IT Operations-4.4	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
IT Operations	RBI IT Framework 4.4.b	MIS For Top Management-4.4	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
IS Audit	RBI IT Framework 5	Policy for Information System Audit (IS Audit)-5	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
IS Audit	RBI IT Framework 5	Policy for Information System Audit (IS Audit)-5	Internet-facing virtual machines should be protected with network security groups	3.0.0
IS Audit	RBI IT Framework 5	Policy for Information System Audit (IS Audit)-5	IP Forwarding on your virtual machine should be disabled	3.0.0
IS Audit	RBI IT Framework 5	Policy for Information System Audit (IS Audit)-5	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
IS Audit	RBI IT Framework 5.2	Coverage-5.2	Azure Backup should be enabled for Virtual Machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Business Continuity Planning	RBI IT Framework 6	Business Continuity Planning (BCP) and Disaster Recovery-6	Audit virtual machines without disaster recovery configured	1.0.0
Business Continuity Planning	RBI IT Framework 6	Business Continuity Planning (BCP) and Disaster Recovery-6	Azure Backup should be enabled for Virtual Machines	3.0.0
Business Continuity Planning	RBI IT Framework 6.2	Recovery strategy / Contingency Plan-6.2	Audit virtual machines without disaster recovery configured	1.0.0
Business Continuity Planning	RBI IT Framework 6.2	Recovery strategy / Contingency Plan-6.2	Azure Backup should be enabled for Virtual Machines	3.0.0
Business Continuity Planning	RBI IT Framework 6.3	Recovery strategy / Contingency Plan-6.3	Azure Backup should be enabled for Virtual Machines	3.0.0
Business Continuity Planning	RBI IT Framework 6.4	Recovery strategy / Contingency Plan-6.4	Audit virtual machines without disaster recovery configured	1.0.0

## RMIT Malaysia

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - RMIT Malaysia](#). For more information about this compliance standard, see [RMIT Malaysia](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Datacenter Operations	RMiT 10.27	Datacenter Operations - 10.27	Deploy - Configure Log Analytics extension to be enabled on Windows virtual machine scale sets	3.0.1
Datacenter Operations	RMiT 10.27	Datacenter Operations - 10.27	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
Datacenter Operations	RMiT 10.30	Datacenter Operations - 10.30	Azure Backup should be enabled for Virtual Machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Network Resilience	RMiT 10.33	Network Resilience - 10.33	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	Configure managed disks to disable public network access	2.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	Internet-facing virtual machines should be protected with network security groups	3.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	IP Forwarding on your virtual machine should be disabled	3.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	Managed disks should disable public network access	2.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Network Resilience	RMiT 10.33	Network Resilience - 10.33	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Network Resilience	RMiT 10.35	Network Resilience - 10.35	Deploy - Configure Log Analytics extension to be enabled on Windows virtual machine scale sets	3.0.1
Cloud Services	RMiT 10.49	Cloud Services - 10.49	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Cloud Services	RMiT 10.49	Cloud Services - 10.49	Management ports should be closed on your virtual machines	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Cloud Services	RMiT 10.51	Cloud Services - 10.51	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
Cloud Services	RMiT 10.51	Cloud Services - 10.51	Audit virtual machines without disaster recovery configured	1.0.0
Cloud Services	RMiT 10.53	Cloud Services - 10.53	Managed disks should use a specific set of disk encryption sets for the customer-managed key encryption	2.0.0
Cloud Services	RMiT 10.53	Cloud Services - 10.53	OS and data disks should be encrypted with a customer-managed key	3.0.0
Access Control	RMiT 10.54	Access Control - 10.54	Guest Configuration extension should be installed on your machines	1.0.2
Access Control	RMiT 10.54	Access Control - 10.54	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0
Access Control	RMiT 10.54	Access Control - 10.54	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Access Control	RMiT 10.61	Access Control - 10.61	Guest Configuration extension should be installed on your machines	1.0.2
Access Control	RMiT 10.61	Access Control - 10.61	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Access Control	RMiT 10.61	Access Control - 10.61	Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity	1.0.1
Patch and End-of-Life System Management	RMiT 10.63	Patch and End-of-Life System Management - 10.63	Microsoft Antimalware for Azure should be configured to automatically update protection signatures	1.0.0
Patch and End-of-Life System Management	RMiT 10.63	Patch and End-of-Life System Management - 10.63	System updates on virtual machine scale sets should be installed	3.0.0
Patch and End-of-Life System Management	RMiT 10.65	Patch and End-of-Life System Management - 10.65	System updates should be installed on your machines	4.0.0
Patch and End-of-Life System Management	RMiT 10.65	Patch and End-of-Life System Management - 10.65	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Security of Digital Services	RMiT 10.66	Security of Digital Services - 10.66	Deploy - Configure Log Analytics extension to be enabled on Windows virtual machines	3.0.1
Security of Digital Services	RMiT 10.66	Security of Digital Services - 10.66	Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images	2.0.1
Security of Digital Services	RMiT 10.66	Security of Digital Services - 10.66	The Log Analytics extension should be installed on Virtual Machine Scale Sets	1.0.1
Security of Digital Services	RMiT 10.66	Security of Digital Services - 10.66	Virtual machines should have the Log Analytics extension installed	1.0.1
Data Loss Prevention (DLP)	RMiT 11.15	Data Loss Prevention (DLP) - 11.15	Configure managed disks to disable public network access	2.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Data Loss Prevention (DLP)	RMiT 11.15	Data Loss Prevention (DLP) - 11.15	Managed disks should disable public network access	2.0.0
Data Loss Prevention (DLP)	RMiT 11.15	Data Loss Prevention (DLP) - 11.15	Managed disks should use a specific set of disk encryption sets for the customer-managed key encryption	2.0.0
Security Operations Centre (SOC)	RMiT 11.17	Security Operations Centre (SOC) - 11.17	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Security Operations Centre (SOC)	RMiT 11.17	Security Operations Centre (SOC) - 11.17	Allowlist rules in your adaptive application control policy should be updated	3.0.0
Security Operations Centre (SOC)	RMiT 11.17	Security Operations Centre (SOC) - 11.17	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Security Operations Centre (SOC)	RMiT 11.17	Security Operations Centre (SOC) - 11.17	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Security Operations Centre (SOC)	RMiT 11.18	Security Operations Centre (SOC) - 11.18	Resource logs in Virtual Machine Scale Sets should be enabled	2.1.0
Cyber Risk Management	RMiT 11.2	Cyber Risk Management - 11.2	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
Cyber Risk Management	RMiT 11.2	Cyber Risk Management - 11.2	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
Security Operations Centre (SOC)	RMiT 11.20	Security Operations Centre (SOC) - 11.20	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
Security Operations Centre (SOC)	RMiT 11.20	Security Operations Centre (SOC) - 11.20	Virtual machines and virtual machine scale sets should have encryption at host enabled	1.0.0
Cyber Risk Management	RMiT 11.4	Cyber Risk Management - 11.4	Configure backup on virtual machines without a given tag to an existing recovery services vault in the same location	9.0.0
Cyber Risk Management	RMiT 11.4	Cyber Risk Management - 11.4	Configure backup on virtual machines without a given tag to an existing recovery services vault in the same location	9.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Cyber Risk Management	RMiT 11.4	Cyber Risk Management - 11.4	Only approved VM extensions should be installed	1.0.0
Cyber Risk Management	RMiT 11.4	Cyber Risk Management - 11.4	Only approved VM extensions should be installed	1.0.0
Cybersecurity Operations	RMiT 11.8	Cybersecurity Operations - 11.8	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.2	Control Measures on Cybersecurity - Appendix 5.2	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.2	Control Measures on Cybersecurity - Appendix 5.2	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Internet-facing virtual machines should be protected with network security groups	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	IP Forwarding on your virtual machine should be disabled	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring	1.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring	1.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Microsoft IaaSAntimalware extension should be deployed on Windows servers	1.1.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Non-internet-facing virtual machines should be protected with network security groups	3.0.0
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Control Measures on Cybersecurity	RMiT Appendix 5.7	Control Measures on Cybersecurity - Appendix 5.7	Vulnerabilities in container security configurations should be remediated	3.0.0

## UK OFFICIAL and UK NHS

To review how the available Azure Policy built-ins for all Azure services map to this compliance standard, see [Azure Policy Regulatory Compliance - UK OFFICIAL and UK NHS](#). For more information about this compliance standard, see [UK OFFICIAL](#).

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY (AZURE PORTAL)	POLICY VERSION (GITHUB)
Data in transit protection	1	Data in transit protection	Windows web servers should be configured to use secure communication protocols	4.0.0
Identity and authentication	10	Identity and authentication	Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities	4.0.0

DOMAIN	CONTROL ID	CONTROL TITLE	POLICY	POLICY VERSION
Identity and authentication	10	Identity and authentication	Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity	4.0.0
Identity and authentication	10	Identity and authentication	Audit Linux machines that allow remote connections from accounts without passwords	3.0.0
Identity and authentication	10	Identity and authentication	Audit Linux machines that do not have the passwd file permissions set to 0644	3.0.0
Identity and authentication	10	Identity and authentication	Audit Linux machines that have accounts without passwords	3.0.0
Identity and authentication	10	Identity and authentication	Audit VMs that do not use managed disks	1.0.0
Identity and authentication	10	Identity and authentication	Audit Windows machines that allow re-use of the previous 24 passwords	2.0.0
Identity and authentication	10	Identity and authentication	Audit Windows machines that do not have a maximum password age of 70 days	2.0.0
Identity and authentication	10	Identity and authentication	Audit Windows machines that do not have a minimum password age of 1 day	2.0.0
Identity and authentication	10	Identity and authentication	Audit Windows machines that do not have the password complexity setting enabled	2.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Identity and authentication	10	Identity and authentication	Audit Windows machines that do not restrict the minimum password length to 14 characters	2.0.0
Identity and authentication	10	Identity and authentication	Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	3.0.0
Identity and authentication	10	Identity and authentication	Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs	1.2.0
Identity and authentication	10	Identity and authentication	Virtual machines should be migrated to new Azure Resource Manager resources	1.0.0
External interface protection	11	External interface protection	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
External interface protection	11	External interface protection	Adaptive network hardening recommendations should be applied on internet facing virtual machines	3.0.0
External interface protection	11	External interface protection	All network ports should be restricted on network security groups associated to your virtual machine	3.0.0
External interface protection	11	External interface protection	Endpoint protection solution should be installed on virtual machine scale sets	3.0.0
External interface protection	11	External interface protection	Management ports of virtual machines should be protected with just-in-time network access control	3.0.0

Domain	Control ID	Control Title	Policy	Policy Version
Asset protection and resilience	2.3	Data at rest protection	Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources	2.0.3
Operational security	5.2	Vulnerability management	A vulnerability assessment solution should be enabled on your virtual machines	3.0.0
Operational security	5.2	Vulnerability management	Monitor missing Endpoint Protection in Azure Security Center	3.0.0
Operational security	5.2	Vulnerability management	System updates on virtual machine scale sets should be installed	3.0.0
Operational security	5.2	Vulnerability management	System updates should be installed on your machines	4.0.0
Operational security	5.2	Vulnerability management	Vulnerabilities in security configuration on your machines should be remediated	3.0.0
Operational security	5.2	Vulnerability management	Vulnerabilities in security configuration on your virtual machine scale sets should be remediated	3.0.0
Operational security	5.3	Protective Monitoring	Adaptive application controls for defining safe applications should be enabled on your machines	3.0.0
Operational security	5.3	Protective Monitoring	Audit virtual machines without disaster recovery configured	1.0.0

## Next steps

- Learn more about [Azure Policy Regulatory Compliance](#).
- See the built-ins on the [Azure Policy GitHub repo](#).

# Azure Policy built-in definitions for Azure Virtual Machines

9/21/2022 • 68 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This page is an index of [Azure Policy](#) built-in policy definitions for Azure Virtual Machines. For additional Azure Policy built-ins for other services, see [Azure Policy built-in definitions](#).

The name of each built-in policy definition links to the policy definition in the Azure portal. Use the link in the **Version** column to view the source on the [Azure Policy GitHub repo](#).

## Microsoft.Compute

NAME (AZURE PORTAL)	DESCRIPTION	EFFECT(S)	VERSION (GITHUB)
[Preview]: [Preview]: Add user-assigned managed identity to enable Guest Configuration assignments on virtual machines	This policy adds a user-assigned managed identity to virtual machines hosted in Azure that are supported by Guest Configuration. A user-assigned managed identity is a prerequisite for all Guest Configuration assignments and must be added to machines before using any Guest Configuration policy definitions. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, DeployIfNotExists, Disabled	<a href="#">2.0.0-preview</a>
[Preview]: [Preview]: Assign Built-In User-Assigned Managed Identity to Virtual Machine Scale Sets	Create and assign a built-in user-assigned managed identity or assign a pre-created user-assigned managed identity at scale to virtual machine scale sets. For more detailed documentation, visit <a href="http://aka.ms/managedidentitypolicy">aka.ms/managedidentitypolicy</a> .	AuditIfNotExists, DeployIfNotExists, Disabled	<a href="#">1.0.2-preview</a>
[Preview]: [Preview]: Assign Built-In User-Assigned Managed Identity to Virtual Machines	Create and assign a built-in user-assigned managed identity or assign a pre-created user-assigned managed identity at scale to virtual machines. For more detailed documentation, visit <a href="http://aka.ms/managedidentitypolicy">aka.ms/managedidentitypolicy</a> .	AuditIfNotExists, DeployIfNotExists, Disabled	<a href="#">1.0.2-preview</a>

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Azure Security agent should be installed on your Linux virtual machine scale sets	Install the Azure Security agent on your Linux virtual machine scale sets in order to monitor your machines for security configurations and vulnerabilities. Results of the assessments can be seen and managed in Azure Security Center.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Azure Security agent should be installed on your Linux virtual machines	Install the Azure Security agent on your Linux virtual machines in order to monitor your machines for security configurations and vulnerabilities. Results of the assessments can be seen and managed in Azure Security Center.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Azure Security agent should be installed on your Windows virtual machine scale sets	Install the Azure Security agent on your Windows virtual machine scale sets in order to monitor your machines for security configurations and vulnerabilities. Results of the assessments can be seen and managed in Azure Security Center.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Azure Security agent should be installed on your Windows virtual machines	Install the Azure Security agent on your Windows virtual machines in order to monitor your machines for security configurations and vulnerabilities. Results of the assessments can be seen and managed in Azure Security Center.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: ChangeTracking extension should be installed on your Linux virtual machine	Install ChangeTracking Extension on Linux virtual machines to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitoring Agent.	AuditIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: ChangeTracking extension should be installed on your Linux virtual machine scale sets	Install ChangeTracking Extension on Linux virtual machine scale sets to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitoring Agent.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: ChangeTracking extension should be installed on your Windows virtual machine	Install ChangeTracking Extension on Windows virtual machines to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitoring Agent.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: ChangeTracking extension should be installed on your Windows virtual machine scale sets	Install ChangeTracking Extension on Windows virtual machine scale sets to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitoring Agent.	AuditIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure Association to link virtual machines to default Microsoft Defender for Cloud Data Collection Rule	Configure machines to automatically create an association with the default data collection rule for Microsoft Defender for Cloud. Deleting this association will break the detection of security vulnerabilities for this virtual machine. Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	2.1.1-preview
[Preview]: [Preview]: Configure Association to link virtual machines to user-defined Microsoft Defender for Cloud Data Collection Rule	Configure machines to automatically create an association with the user-defined data collection rule for Microsoft Defender for Cloud. Deleting this association will break the detection of security vulnerabilities for this virtual machine. Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Configure Azure Defender for SQL agent on virtual machine	Configure Windows machines to automatically install the Azure Defender for SQL agent where the Azure Monitor Agent is installed. Security Center collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Creates a resource group and Log Analytics workspace in the same region as the machine. Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure ChangeTracking Extension for Linux virtual machine scale sets	Configure Linux virtual machine scale sets to automatically install the ChangeTracking Extension to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitor Agent.	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Configure ChangeTracking Extension for Linux virtual machines	Configure Linux virtual machines to automatically install the ChangeTracking Extension to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitor Agent.	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Configure ChangeTracking Extension for Windows virtual machine scale sets	Configure Windows virtual machine scale sets to automatically install the ChangeTracking Extension to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitor Agent.	DeployIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure ChangeTracking Extension for Windows virtual machines	Configure Windows virtual machines to automatically install the ChangeTracking Extension to enable File Integrity Monitoring(FIM) in Azure Security Center. FIM examines operating system files, Windows registries, application software, Linux system files, and more, for changes that might indicate an attack. The extension can be installed in virtual machines and locations supported by Azure Monitor Agent.	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Configure machines to create the Microsoft Defender for Cloud user-defined pipeline using Azure Monitor Agent	Configure machines to create the Microsoft Defender for Cloud user-defined pipeline using Azure Monitor Agent. Microsoft Defender for Cloud collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Use the user-provided Log Analytics workspace to store audit records. Creates a resource group and a Data Collection Rule in the same region as the user-provided Log Analytics workspace. Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	1.2.0-preview
[Preview]: [Preview]: Configure periodic checking for missing system updates on azure virtual machines	Configure auto-assessment (every 24 hours) for OS updates on native Azure virtual machines. You can control the scope of assignment according to machine subscription, resource group, location or tag. Learn more about this for Windows: <a href="https://aka.ms/computevm-windowspatchassessmentmode">https://aka.ms/computevm-windowspatchassessmentmode</a> , for Linux: <a href="https://aka.ms/computevm-linuxpatchassessmentmode">https://aka.ms/computevm-linuxpatchassessmentmode</a> .	modify	2.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure supported Linux virtual machine scale sets to automatically install the Azure Security agent	Configure supported Linux virtual machine scale sets to automatically install the Azure Security agent. Security Center collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Configure supported Linux virtual machine scale sets to automatically install the Guest Attestation extension	Configure supported Linux virtual machines scale sets to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	5.0.0-preview
[Preview]: [Preview]: Configure supported Linux virtual machines to automatically enable Secure Boot	Configure supported Linux virtual machines to automatically enable Secure Boot to mitigate against malicious and unauthorized changes to the boot chain. Once enabled, only trusted bootloaders, kernel and kernel drivers will be allowed to run.	DeployIfNotExists, Disabled	5.0.0-preview
[Preview]: [Preview]: Configure supported Linux virtual machines to automatically install the Azure Security agent	Configure supported Linux virtual machines to automatically install the Azure Security agent. Security Center collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	6.0.0-preview
[Preview]: [Preview]: Configure supported Linux virtual machines to automatically install the Guest Attestation extension	Configure supported Linux virtual machines to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	6.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure supported virtual machines to automatically enable vTPM	Configure supported virtual machines to automatically enable vTPM to facilitate Measured Boot and other OS security features that require a TPM. Once enabled, vTPM can be used to attest boot integrity.	DeployIfNotExists, Disabled	2.0.0-preview
[Preview]: [Preview]: Configure supported Windows machines to automatically install the Azure Security agent	Configure supported Windows machines to automatically install the Azure Security agent. Security Center collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	4.0.0-preview
[Preview]: [Preview]: Configure supported Windows virtual machine scale sets to automatically install the Azure Security agent	Configure supported Windows virtual machine scale sets to automatically install the Azure Security agent. Security Center collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Target Windows virtual machine scale sets must be in a supported location.	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Configure supported Windows virtual machine scale sets to automatically install the Guest Attestation extension	Configure supported Windows virtual machines scale sets to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	3.0.0-preview
[Preview]: [Preview]: Configure supported Windows virtual machines to automatically enable Secure Boot	Configure supported Windows virtual machines to automatically enable Secure Boot to mitigate against malicious and unauthorized changes to the boot chain. Once enabled, only trusted bootloaders, kernel and kernel drivers will be allowed to run.	DeployIfNotExists, Disabled	3.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure supported Windows virtual machines to automatically install the Guest Attestation extension	Configure supported Windows virtual machines to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	4.0.0-preview
[Preview]: [Preview]: Configure system-assigned managed identity to enable Azure Monitor assignments on VMs	Configure system-assigned managed identity to virtual machines hosted in Azure that are supported by Azure Monitor and do not have a system-assigned managed identity. A system-assigned managed identity is a prerequisite for all Azure Monitor assignments and must be added to machines before using any Azure Monitor extension. Target virtual machines must be in a supported location.	Modify, Disabled	5.0.0-preview
[Preview]: [Preview]: Configure virtual machines to create the default Microsoft Defender for Cloud pipeline using Azure Monitor Agent	Configure virtual machines to create the default Microsoft Defender for Cloud pipeline using Azure Monitor Agent. Microsoft Defender for Cloud collects events from the agent and uses them to provide security alerts and tailored hardening tasks (recommendations). Create a resource group, a Data Collection Rule and Log Analytics workspace in the same region as the machine to store audit records. Target virtual machines must be in a supported location.	DeployIfNotExists, Disabled	5.2.0-preview
[Preview]: [Preview]: Configure VMs created with Shared Image Gallery images to install the Guest Attestation extension	Configure virtual machines created with Shared Image Gallery images to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	2.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Configure VMSS created with Shared Image Gallery images to install the Guest Attestation extension	Configure VMSS created with Shared Image Gallery images to automatically install the Guest Attestation extension to allow Azure Security Center to proactively attest and monitor the boot integrity. Boot integrity is attested via Remote Attestation.	DeployIfNotExists, Disabled	2.0.0-preview
[Preview]: [Preview]: Deploy a VMInsights Data Collection Rule and Data Collection Rule Association for all the VMs in the Resource Group	Deploy a Data Collection Rule for VMInsights and deploy Data Collection Rule Association for all the VMs in the Resource Group. The policy asks if enabling of Processes and Dependencies is required and accordingly creates the DCR.	DeployIfNotExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Deploy a VMInsights Data Collection Rule and Data Collection Rule Association for all the VMSS in the Resource Group	Deploy a Data Collection Rule for VMInsights and deploy Data Collection Rule Association for all the VMSSs in the Resource Group. The policy asks if enabling of Processes and Dependencies is required and accordingly creates the DCR.	DeployIfNotExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Deploy Dependency agent for Linux virtual machine scale sets with Azure Monitoring Agent settings	Deploy Dependency agent for Linux virtual machine scale sets with Azure Monitoring Agent settings if the VM Image (OS) is in the list defined and the agent is not installed. Note: if your scale set upgradePolicy is set to Manual, you need to apply the extension to the all virtual machines in the set by calling upgrade on them. In CLI this would be az vmss update-instances.	DeployIfNotExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Deploy Dependency agent for Linux virtual machines with Azure Monitoring Agent settings	Deploy Dependency agent for Linux virtual machines with Azure Monitoring Agent settings if the VM Image (OS) is in the list defined and the agent is not installed.	DeployIfNotExists, Disabled	1.1.1-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Deploy Dependency agent to be enabled on Windows virtual machine scale sets with Azure Monitoring Agent settings	Deploy Dependency agent for Windows virtual machine scale sets with Azure Monitoring Agent settings if the virtual machine image is in the list defined and the agent is not installed. If your scale set upgradePolicy is set to Manual, you need to apply the extension to all the virtual machines in the set by updating them.	DeployIfExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Deploy Dependency agent to be enabled on Windows virtual machines with Azure Monitoring Agent settings	Deploy Dependency agent for Windows virtual machines with Azure Monitoring Agent settings if the virtual machine image is in the list defined and the agent is not installed.	DeployIfExists, Disabled	1.1.1-preview
[Preview]: [Preview]: Deploy Microsoft Defender for Endpoint agent on Linux virtual machines	Deploys Microsoft Defender for Endpoint agent on applicable Linux VM images.	DeployIfExists, AuditIfExists, Disabled	2.0.1-preview
[Preview]: [Preview]: Deploy Microsoft Defender for Endpoint agent on Windows virtual machines	Deploys Microsoft Defender for Endpoint on applicable Windows VM images.	DeployIfExists, AuditIfExists, Disabled	2.0.1-preview
[Preview]: [Preview]: Guest Attestation extension should be installed on supported Linux virtual machines	Install Guest Attestation extension on supported Linux virtual machines to allow Azure Security Center to proactively attest and monitor the boot integrity. Once installed, boot integrity will be attested via Remote Attestation. This assessment only applies to trusted launch enabled Linux virtual machines.	AuditIfExists, Disabled	5.0.0-preview

Name	Description	Effect(s)	Version
[Preview]: [Preview]: Guest Attestation extension should be installed on supported Linux virtual machines scale sets	Install Guest Attestation extension on supported Linux virtual machines scale sets to allow Azure Security Center to proactively attest and monitor the boot integrity. Once installed, boot integrity will be attested via Remote Attestation. This assessment only applies to trusted launch enabled Linux virtual machine scale sets.	AuditIfNotExists, Disabled	4.0.0-preview
[Preview]: [Preview]: Guest Attestation extension should be installed on supported Windows virtual machines	Install Guest Attestation extension on supported virtual machines to allow Azure Security Center to proactively attest and monitor the boot integrity. Once installed, boot integrity will be attested via Remote Attestation. This assessment only applies to trusted launch enabled virtual machines.	AuditIfNotExists, Disabled	3.0.0-preview
[Preview]: [Preview]: Guest Attestation extension should be installed on supported Windows virtual machines scale sets	Install Guest Attestation extension on supported virtual machines scale sets to allow Azure Security Center to proactively attest and monitor the boot integrity. Once installed, boot integrity will be attested via Remote Attestation. This assessment only applies to trusted launch enabled virtual machine scale sets.	AuditIfNotExists, Disabled	2.0.0-preview
[Preview]: [Preview]: Linux machines should meet requirements for the Azure security baseline for Docker hosts	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . The machine is not configured correctly for one of the recommendations in the Azure security baseline for Docker hosts.	AuditIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Linux machines with OMI installed should have version 1.6.8-1 or later	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>. Due to a security fix included in version 1.6.8-1 of the OMI package for Linux, all machines should be updated to the latest release. Upgrade apps/packages that use OMI to resolve the issue. For more information, see <a href="https://aka.ms/omiguidance">https://aka.ms/omiguidance</a>.</p>	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Linux virtual machines should use Secure Boot	<p>To protect against the installation of malware-based rootkits and boot kits, enable Secure Boot on supported Linux virtual machines. Secure Boot ensures that only signed operating systems and drivers will be allowed to run. This assessment only applies to Linux virtual machines that have the Azure Monitor Agent installed.</p>	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Log Analytics Extension should be enabled for listed virtual machine images	<p>Reports virtual machines as non-compliant if the virtual machine image is not in the list defined and the extension is not installed.</p>	AuditIfNotExists, Disabled	2.0.1-preview
[Preview]: [Preview]: Machines should be configured to periodically check for missing system updates	<p>To ensure periodic assessments for missing system updates are triggered automatically every 24 hours, the AssessmentMode property should be set to 'AutomaticByPlatform'. Learn more about AssessmentMode property for Windows: <a href="https://aka.ms/computevm-windowspatchassessmentmode">https://aka.ms/computevm-windowspatchassessmentmode</a>, for Linux: <a href="https://aka.ms/computevm-linuxpatchassessmentmode">https://aka.ms/computevm-linuxpatchassessmentmode</a>.</p>	Audit, Deny, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Machines should have ports closed that might expose attack vectors	Azure's Terms Of Use prohibit the use of Azure services in ways that could damage, disable, overburden, or impair any Microsoft server, or the network. The exposed ports identified by this recommendation need to be closed for your continued security. For each identified port, the recommendation also provides an explanation of the potential threat.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Network traffic data collection agent should be installed on Linux virtual machines	Security Center uses the Microsoft Dependency agent to collect network traffic data from your Azure virtual machines to enable advanced network protection features such as traffic visualization on the network map, network hardening recommendations and specific network threats.	AuditIfNotExists, Disabled	1.0.2-preview
[Preview]: [Preview]: Network traffic data collection agent should be installed on Windows virtual machines	Security Center uses the Microsoft Dependency agent to collect network traffic data from your Azure virtual machines to enable advanced network protection features such as traffic visualization on the network map, network hardening recommendations and specific network threats.	AuditIfNotExists, Disabled	1.0.2-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
[Preview]: [Preview]: Schedule recurring updates using Update Management Center	<p>You can use update management center (private preview) in Azure to save recurring deployment schedules to install operating system updates for your Windows Server and Linux machines in Azure, in on-premises environments, and in other cloud environments connected using Azure Arc-enabled servers. This policy will also change the patch mode for the Azure Virtual Machine to 'AutomaticByPlatform'. See more: <a href="https://aka.ms/umc-scheduled-patching">https://aka.ms/umc-scheduled-patching</a></p>	DeployIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: Secure Boot should be enabled on supported Windows virtual machines	<p>Enable Secure Boot on supported Windows virtual machines to mitigate against malicious and unauthorized changes to the boot chain. Once enabled, only trusted bootloaders, kernel and kernel drivers will be allowed to run. This assessment only applies to trusted launch enabled Windows virtual machines.</p>	Audit, Disabled	3.0.0-preview
[Preview]: [Preview]: System updates should be installed on your machines (powered by Update Center)	<p>Your machines are missing system, security, and critical updates. Software updates often include critical patches to security holes. Such holes are frequently exploited in malware attacks so it's vital to keep your software updated. To install all outstanding patches and secure your machines, follow the remediation steps.</p>	AuditIfExists, Disabled	1.0.0-preview

Name	Description	Effect(s)	Version
[Preview]: [Preview]: Virtual machines guest attestation status should be healthy	Guest attestation is performed by sending a trusted log (TCGLog) to an attestation server. The server uses these logs to determine whether boot components are trustworthy. This assessment is intended to detect compromises of the boot chain which might be the result of a bootkit or rootkit infection. This assessment only applies to Trusted Launch enabled virtual machines that have Guest Attestation extension installed.	AuditIfNotExists, Disabled	1.0.0-preview
[Preview]: [Preview]: vTPM should be enabled on supported virtual machines	Enable virtual TPM device on supported virtual machines to facilitate Measured Boot and other OS security features that require a TPM. Once enabled, vTPM can be used to attest boot integrity. This assessment only applies to trusted launch enabled virtual machines.	Audit, Disabled	2.0.0-preview
[Preview]: [Preview]: Windows machines should meet STIG compliance requirements for Azure compute	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if the machine is not configured correctly for one of the recommendations in STIG compliance requirements for Azure compute. DISA (Defense Information Systems Agency) provides technical guides STIG (Security Technical Implementation Guide) to secure compute OS as required by Department of Defense (DoD). For more details, <a href="https://public.cyber.mil/stigs/">https://public.cyber.mil/stigs/</a> .	AuditIfNotExists, Disabled	1.0.0-preview

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	Audits virtual machines to detect whether they are running a supported vulnerability assessment solution. A core component of every cyber risk and security program is the identification and analysis of vulnerabilities. Azure Security Center's standard pricing tier includes vulnerability scanning for your virtual machines at no extra cost. Additionally, Security Center can automatically deploy this tool for you.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	Enable application controls to define the list of known-safe applications running on your machines, and alert you when other applications run. This helps harden your machines against malware. To simplify the process of configuring and maintaining your rules, Security Center uses machine learning to analyze the applications running on each machine and suggest the list of known-safe applications.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Adaptive network hardening recommendations should be applied on internet facing virtual machines</a>	Azure Security Center analyzes the traffic patterns of Internet facing virtual machines and provides Network Security Group rule recommendations that reduce the potential attack surface	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on virtual machines with no identities</a>	This policy adds a system-assigned managed identity to virtual machines hosted in Azure that are supported by Guest Configuration but do not have any managed identities. A system-assigned managed identity is a prerequisite for all Guest Configuration assignments and must be added to machines before using any Guest Configuration policy definitions. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	modify	4.0.0
<a href="#">Add system-assigned managed identity to enable Guest Configuration assignments on VMs with a user-assigned identity</a>	This policy adds a system-assigned managed identity to virtual machines hosted in Azure that are supported by Guest Configuration and have at least one user-assigned identity but do not have a system-assigned managed identity. A system-assigned managed identity is a prerequisite for all Guest Configuration assignments and must be added to machines before using any Guest Configuration policy definitions. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	modify	4.0.0
<a href="#">All network ports should be restricted on network security groups associated to your virtual machine</a>	Azure Security Center has identified some of your network security groups' inbound rules to be too permissive. Inbound rules should not allow access from 'Any' or 'Internet' ranges. This can potentially enable attackers to target your resources.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Allowed virtual machine size SKUs</a>	This policy enables you to specify a set of virtual machine size SKUs that your organization can deploy.	Deny	1.0.1

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Allowlist rules in your adaptive application control policy should be updated</a>	Monitor for changes in behavior on groups of machines configured for auditing by Azure Security Center's adaptive application controls. Security Center uses machine learning to analyze the running processes on your machines and suggest a list of known-safe applications. These are presented as recommended apps to allow in adaptive application control policies.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Audit Linux machines that allow remote connections from accounts without passwords</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if Linux machines that allow remote connections from accounts without passwords	AuditIfNotExists, Disabled	3.0.0
<a href="#">Audit Linux machines that do not have the passwd file permissions set to 0644</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if Linux machines that do not have the passwd file permissions set to 0644	AuditIfNotExists, Disabled	3.0.0
<a href="#">Audit Linux machines that don't have the specified applications installed</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if the Chef InSpec resource indicates that one or more of the packages provided by the parameter are not installed.	AuditIfNotExists, Disabled	4.0.0
<a href="#">Audit Linux machines that have accounts without passwords</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if Linux machines that have accounts without passwords	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Linux machines that have the specified applications installed</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the Chef InSpec resource indicates that one or more of the packages provided by the parameter are installed.</p>	AuditIfNotExists, Disabled	4.0.0
<a href="#">Audit virtual machines without disaster recovery configured</a>	<p>Audit virtual machines which do not have disaster recovery configured. To learn more about disaster recovery, visit <a href="https://aka.ms/asr-doc">https://aka.ms/asr-doc</a>.</p>	auditIfExists	1.0.0
<a href="#">Audit VMs that do not use managed disks</a>	This policy audits VMs that do not use managed disks	audit	1.0.0
<a href="#">Audit Windows machines missing any of specified members in the Administrators group</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the local Administrators group does not contain one or more members that are listed in the policy parameter.</p>	auditIfExists	2.0.0
<a href="#">Audit Windows machines network connectivity</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if a network connection status to an IP and TCP port does not match the policy parameter.</p>	auditIfExists	2.0.0
<a href="#">Audit Windows machines on which the DSC configuration is not compliant</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the Windows PowerShell command Get-DSCConfigurationStatus returns that the DSC configuration for the machine is not compliant.</p>	auditIfExists	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows machines on which the Log Analytics agent is not connected as expected</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the agent is not installed, or if it is installed but the COM object AgentConfigManager.Mgmt SvcCfg returns that it is registered to a workspace other than the ID specified in the policy parameter.</p>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines on which the specified services are not installed and 'Running'</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if result of the Windows PowerShell command Get-Service do not include the service name with matching status as specified by the policy parameter.</p>	auditIfNotExists	3.0.0
<a href="#">Audit Windows machines on which Windows Serial Console is not enabled</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the machine does not have the Serial Console software installed or if the EMS port number or baud rate are not configured with the same values as the policy parameters.</p>	auditIfNotExists	3.0.0
<a href="#">Audit Windows machines that allow re-use of the previous 24 passwords</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that allow re-use of the previous 24 passwords</p>	AuditIfNotExists, Disabled	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows machines that are not joined to the specified domain</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the value of the Domain property in WMI class win32_computerSystem does not match the value in the policy parameter.</p>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines that are not set to the specified time zone</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the value of the property StandardName in WMI class Win32_TimeZone does not match the selected time zone for the policy parameter.</p>	auditIfNotExists	3.0.0
<a href="#">Audit Windows machines that contain certificates expiring within the specified number of days</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if certificates in the specified store have an expiration date out of range for the number of days given as parameter. The policy also provides the option to only check for specific certificates or exclude specific certificates, and whether to report on expired certificates.</p>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines that do not contain the specified certificates in Trusted Root</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the machine Trusted Root certificate store (Cert:\LocalMachine\Root) does not contain one or more of the certificates listed by the policy parameter.</p>	auditIfNotExists	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows machines that do not have a maximum password age of 70 days</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that do not have a maximum password age of 70 days</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Audit Windows machines that do not have a minimum password age of 1 day</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that do not have a minimum password age of 1 day</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Audit Windows machines that do not have the password complexity setting enabled</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that do not have the password complexity setting enabled</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Audit Windows machines that do not have the specified Windows PowerShell execution policy</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the Windows PowerShell command Get-ExecutionPolicy returns a value other than what was selected in the policy parameter.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Audit Windows machines that do not have the specified Windows PowerShell modules installed</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if a module isn't available in a location specified by the environment variable PSModulePath.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows machines that do not restrict the minimum password length to 14 characters</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that do not restrict the minimum password length to 14 characters</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Audit Windows machines that do not store passwords using reversible encryption</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if Windows machines that do not store passwords using reversible encryption</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Audit Windows machines that don't have the specified applications installed</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the application name is not found in any of the following registry paths:</p> <ul style="list-style-type: none"> <li>HKLM:SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,</li> <li>HKLM:SOFTWARE\Wow6432node\Microsoft\Windows\CurrentVersion\Uninstall,</li> <li>HKCU:Software\Microsoft\Windows\CurrentVersion\Uninstall.</li> </ul>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines that have extra accounts in the Administrators group</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the local Administrators group contains members that are not listed in the policy parameter.</p>	auditIfNotExists	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows machines that have not restarted within the specified number of days</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the WMI property LastBootUpTime in class Win32_Operatingsystem is outside the range of days provided by the policy parameter.</p>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines that have the specified applications installed</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the application name is found in any of the following registry paths:</p> <ul style="list-style-type: none"> <li>HKLM:SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall,</li> <li>HKLM:SOFTWARE\Wow6432node\Microsoft\Windows\CurrentVersion\Uninstall,</li> <li>HKCU:Software\Microsoft\Windows\CurrentVersion\Uninstall.</li> </ul>	auditIfNotExists	2.0.0
<a href="#">Audit Windows machines that have the specified members in the Administrators group</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the local Administrators group contains one or more of the members listed in the policy parameter.</p>	auditIfNotExists	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Audit Windows VMs with a pending reboot</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the machine is pending reboot for any of the following reasons: component based servicing, Windows Update, pending file rename, pending computer rename, configuration manager pending reboot. Each detection has a unique registry path.</p>	auditIfNotExists	2.0.0
<a href="#">Authentication to Linux machines should require SSH keys</a>	<p>Although SSH itself provides an encrypted connection, using passwords with SSH still leaves the VM vulnerable to brute-force attacks. The most secure option for authenticating to an Azure Linux virtual machine over SSH is with a public-private key pair, also known as SSH keys. Learn more: <a href="https://docs.microsoft.com/azure/virtual-machines/linux/create-ssh-keys-detailed">https://docs.microsoft.com/azure/virtual-machines/linux/create-ssh-keys-detailed</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Azure Backup should be enabled for Virtual Machines</a>	<p>Ensure protection of your Azure Virtual Machines by enabling Azure Backup. Azure Backup is a secure and cost effective data protection solution for Azure.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Cloud Services (extended support) role instances should be configured securely</a>	<p>Protect your Cloud Service (extended support) role instances from attacks by ensuring they are not exposed to any OS vulnerabilities.</p>	AuditIfNotExists, Disabled	1.0.0
<a href="#">Cloud Services (extended support) role instances should have an endpoint protection solution installed</a>	<p>Protect your Cloud Services (extended support) role instances from threats and vulnerabilities by ensuring an endpoint protection solution is installed on them.</p>	AuditIfNotExists, Disabled	1.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Cloud Services (extended support) role instances should have system updates installed</a>	Secure your Cloud Services (extended support) role instances by ensuring the latest security and critical updates are installed on them.	AuditIfNotExists, Disabled	1.0.0
<a href="#">Configure backup on virtual machines with a given tag to a new recovery services vault with a default policy</a>	Enforce backup for all virtual machines by deploying a recovery services vault in the same location and resource group as the virtual machine. Doing this is useful when different application teams in your organization are allocated separate resource groups and need to manage their own backups and restores. You can optionally include virtual machines containing a specified tag to control the scope of assignment. See <a href="https://aka.ms/AzureVMApCentricBackupIncludeTag">https://aka.ms/AzureVMApCentricBackupIncludeTag</a> .	auditIfNotExists, AuditIfNotExists, deployIfNotExists, DeployIfNotExists, disabled, Disabled	9.0.0
<a href="#">Configure backup on virtual machines with a given tag to an existing recovery services vault in the same location</a>	Enforce backup for all virtual machines by backing them up to an existing central recovery services vault in the same location and subscription as the virtual machine. Doing this is useful when there is a central team in your organization managing backups for all resources in a subscription. You can optionally include virtual machines containing a specified tag to control the scope of assignment. See <a href="https://aka.ms/AzureVMCentralBackupIncludeTag">https://aka.ms/AzureVMCentralBackupIncludeTag</a> .	auditIfNotExists, AuditIfNotExists, deployIfNotExists, DeployIfNotExists, disabled, Disabled	9.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Configure backup on virtual machines without a given tag to a new recovery services vault with a default policy</a>	<p>Enforce backup for all virtual machines by deploying a recovery services vault in the same location and resource group as the virtual machine.</p> <p>Doing this is useful when different application teams in your organization are allocated separate resource groups and need to manage their own backups and restores. You can optionally exclude virtual machines containing a specified tag to control the scope of assignment. See <a href="https://aka.ms/AzureVMAp_pCentricBackupExcludeTag">https://aka.ms/AzureVMAp_pCentricBackupExcludeTag</a>.</p>	auditIfNotExists, AuditIfNotExists, deployIfNotExists, DeployIfNotExists, disabled, Disabled	9.0.0
<a href="#">Configure backup on virtual machines without a given tag to an existing recovery services vault in the same location</a>	<p>Enforce backup for all virtual machines by backing them up to an existing central recovery services vault in the same location and subscription as the virtual machine. Doing this is useful when there is a central team in your organization managing backups for all resources in a subscription. You can optionally exclude virtual machines containing a specified tag to control the scope of assignment. See <a href="https://aka.ms/AzureVMCe_ntralBackupExcludeTag">https://aka.ms/AzureVMCe_ntralBackupExcludeTag</a>.</p>	auditIfNotExists, AuditIfNotExists, deployIfNotExists, DeployIfNotExists, disabled, Disabled	9.0.0
<a href="#">Configure disaster recovery on virtual machines by enabling replication via Azure Site Recovery</a>	<p>Virtual machines without disaster recovery configurations are vulnerable to outages and other disruptions. If the virtual machine does not already have disaster recovery configured, this would initiate the same by enabling replication using preset configurations to facilitate business continuity. You can optionally include/exclude virtual machines containing a specified tag to control the scope of assignment. To learn more about disaster recovery, visit <a href="https://aka.ms/asr-doc">https://aka.ms/asr-doc</a>.</p>	DeployIfNotExists, Disabled	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
Configure disk access resources with private endpoints	Private endpoints connect your virtual networks to Azure services without a public IP address at the source or destination. By mapping private endpoints to disk access resources, you can reduce data leakage risks. Learn more about private links at: <a href="https://aka.ms/disksprivatelinksdoc">https://aka.ms/disksprivatelinksdoc</a> .	DeployIfNotExists, Disabled	1.0.0
Configure Linux Machines to be associated with a Data Collection Rule	Deploy Association to link Linux virtual machines, virtual machine scale sets, and Arc machines to the specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfNotExists, Disabled	4.0.0
Configure Linux Virtual Machine Scale Sets to be associated with a Data Collection Rule	Deploy Association to link Linux virtual machine scale sets to the specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfNotExists, Disabled	2.0.0
Configure Linux virtual machine scale sets to run Azure Monitor Agent with system-assigned managed identity-based authentication	Automate the deployment of Azure Monitor Agent extension on your Linux virtual machine scale sets for collecting telemetry data from the guest OS. This policy will install the extension if the OS and region are supported and system-assigned managed identity is enabled, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	2.1.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Configure Linux virtual machine scale sets to run Azure Monitor Agent with user-assigned managed identity-based authentication</a>	Automate the deployment of Azure Monitor Agent extension on your Linux virtual machine scale sets for collecting telemetry data from the guest OS. This policy will install the extension and configure it to use the specified user-assigned managed identity if the OS and region are supported, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	2.1.0
<a href="#">Configure Linux Virtual Machines to be associated with a Data Collection Rule</a>	Deploy Association to link Linux virtual machines to the specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfNotExists, Disabled	2.0.0
<a href="#">Configure Linux virtual machines to run Azure Monitor Agent with system-assigned managed identity-based authentication</a>	Automate the deployment of Azure Monitor Agent extension on your Linux virtual machines for collecting telemetry data from the guest OS. This policy will install the extension if the OS and region are supported and system-assigned managed identity is enabled, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	2.1.0
<a href="#">Configure Linux virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication</a>	Automate the deployment of Azure Monitor Agent extension on your Linux virtual machines for collecting telemetry data from the guest OS. This policy will install the extension and configure it to use the specified user-assigned managed identity if the OS and region are supported, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	2.1.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
Configure machines to receive a vulnerability assessment provider	Azure Defender includes vulnerability scanning for your machines at no extra cost. You don't need a Qualys license or even a Qualys account - everything's handled seamlessly inside Security Center. When you enable this policy, Azure Defender automatically deploys the Qualys vulnerability assessment provider to all supported machines that don't already have it installed.	DeployIfNotExists, Disabled	4.0.0
Configure managed disks to disable public network access	Disable public network access for your managed disk resource so that it's not accessible over the public internet. This can reduce data leakage risks. Learn more at: <a href="https://aka.ms/diskprivatenetworkdoc">https://aka.ms/diskprivatenetworkdoc</a> .	Modify, Disabled	2.0.0
Configure secure communication protocols(TLS 1.1 or TLS 1.2) on Windows servers	Creates a Guest Configuration assignment to configure specified secure protocol version(TLS 1.1 or TLS 1.2) on Windows server	DeployIfNotExists, Disabled	1.0.0
Configure time zone on Windows machines.	This policy creates a Guest Configuration assignment to set specified time zone on Windows virtual machines.	deployIfNotExists	2.0.0
Configure virtual machines to be onboarded to Azure Automanage	Azure Automanage enrolls, configures, and monitors virtual machines with best practice as defined in the Microsoft Cloud Adoption Framework for Azure. Use this policy to apply Automanage to your selected scope.	AuditIfExists, DeployIfExists, Disabled	2.2.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Configure virtual machines to be onboarded to Azure Automanage with Custom Configuration Profile</a>	Azure Automanage enrolls, configures, and monitors virtual machines with best practice as defined in the Microsoft Cloud Adoption Framework for Azure. Use this policy to apply Automanage with your own customized Configuration Profile to your selected scope.	AuditIfNotExists, DeployIfExists, Disabled	1.2.0
<a href="#">Configure Windows Machines to be associated with a Data Collection Rule</a>	Deploy Association to link Windows virtual machines, virtual machine scale sets, and Arc machines to specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfExists, Disabled	2.1.0
<a href="#">Configure Windows Virtual Machine Scale Sets to be associated with a Data Collection Rule</a>	Deploy Association to link Windows virtual machine scale sets to specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfExists, Disabled	1.1.0
<a href="#">Configure Windows virtual machine scale sets to run Azure Monitor Agent using system-assigned managed identity</a>	Automate the deployment of Azure Monitor Agent extension on your Windows virtual machine scale sets for collecting telemetry data from the guest OS. This policy will install the extension if the OS and region are supported and system-assigned managed identity is enabled, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfExists, Disabled	3.1.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Configure Windows virtual machine scale sets to run Azure Monitor Agent with user-assigned managed identity-based authentication</a>	Automate the deployment of Azure Monitor Agent extension on your Windows virtual machine scale sets for collecting telemetry data from the guest OS. This policy will install the extension and configure it to use the specified user-assigned managed identity if the OS and region are supported, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	1.1.0
<a href="#">Configure Windows Virtual Machines to be associated with a Data Collection Rule</a>	Deploy Association to link Windows virtual machines to specified Data Collection Rule. The list of locations and OS images are updated over time as support is increased.	DeployIfNotExists, Disabled	1.1.0
<a href="#">Configure Windows virtual machines to run Azure Monitor Agent using system-assigned managed identity</a>	Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telemetry data from the guest OS. This policy will install the extension if the OS and region are supported and system-assigned managed identity is enabled, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	4.1.0
<a href="#">Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication</a>	Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telemetry data from the guest OS. This policy will install the extension and configure it to use the specified user-assigned managed identity if the OS and region are supported, and skip install otherwise. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	DeployIfNotExists, Disabled	1.1.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
Dependency agent should be enabled for listed virtual machine images	Reports virtual machines as non-compliant if the virtual machine image is not in the list defined and the agent is not installed. The list of OS images is updated over time as support is updated.	AuditIfNotExists, Disabled	2.0.0
Dependency agent should be enabled in virtual machine scale sets for listed virtual machine images	Reports virtual machine scale sets as non-compliant if the virtual machine image is not in the list defined and the agent is not installed. The list of OS images is updated over time as support is updated.	AuditIfNotExists, Disabled	2.0.0
Deploy - Configure Dependency agent to be enabled on Windows virtual machine scale sets	Deploy Dependency agent for Windows virtual machine scale sets if the virtual machine image is in the list defined and the agent is not installed. If your scale set upgradePolicy is set to Manual, you need to apply the extension to all the virtual machines in the set by updating them.	DeployIfExists, Disabled	3.1.0
Deploy - Configure Dependency agent to be enabled on Windows virtual machines	Deploy Dependency agent for Windows virtual machines if the virtual machine image is in the list defined and the agent is not installed.	DeployIfExists, Disabled	3.1.0
Deploy - Configure Log Analytics extension to be enabled on Windows virtual machine scale sets	Deploy Log Analytics extension for Windows virtual machine scale sets if the virtual machine image is in the list defined and the extension is not installed. If your scale set upgradePolicy is set to Manual, you need to apply the extension to all the virtual machine in the set by updating them. Deprecation notice: The Log Analytics agent is on a deprecation path and won't be supported after August 31, 2024. You must migrate to the replacement 'Azure Monitor agent' prior to that date.	DeployIfExists, Disabled	3.0.1

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Deploy - Configure Log Analytics extension to be enabled on Windows virtual machines</a>	Deploy Log Analytics extension for Windows virtual machines if the virtual machine image is in the list defined and the extension is not installed. Deprecation notice: The Log Analytics agent is on a deprecation path and won't be supported after August 31, 2024. You must migrate to the replacement 'Azure Monitor agent' prior to that date.	DeployIfNotExists, Disabled	<a href="#">3.0.1</a>
<a href="#">Deploy default Microsoft IaaSAntimalware extension for Windows Server</a>	This policy deploys a Microsoft IaaSAntimalware extension with a default configuration when a VM is not configured with the antimalware extension.	deployIfExists	<a href="#">1.1.0</a>
<a href="#">Deploy Dependency agent for Linux virtual machine scale sets</a>	Deploy Dependency agent for Linux virtual machine scale sets if the VM Image (OS) is in the list defined and the agent is not installed. Note: if your scale set upgradePolicy is set to Manual, you need to apply the extension to the all virtual machines in the set by calling upgrade on them. In CLI this would be az vmss update-instances.	deployIfExists	<a href="#">4.0.0</a>
<a href="#">Deploy Dependency agent for Linux virtual machines</a>	Deploy Dependency agent for Linux virtual machines if the VM Image (OS) is in the list defined and the agent is not installed.	deployIfExists	<a href="#">4.0.0</a>

NAME	DESCRIPTION	EFFECT(S)	VERSION
Deploy Log Analytics extension for Linux virtual machine scale sets. See deprecation notice below	<p>Deploy Log Analytics extension for Linux virtual machine scale sets if the VM Image (OS) is in the list defined and the extension is not installed. Note: if your scale set upgradePolicy is set to Manual, you need to apply the extension to the all VMs in the set by calling upgrade on them. In CLI this would be az vmss update-instances.</p> <p>Deprecation notice: The Log Analytics agent will not be supported after August 31, 2024. You must migrate to the replacement 'Azure Monitor agent' prior to that date</p>	deployIfNotExists	3.0.0
Deploy Log Analytics extension for Linux VMs. See deprecation notice below	<p>Deploy Log Analytics extension for Linux VMs if the VM Image (OS) is in the list defined and the extension is not installed.</p> <p>Deprecation notice: The Log Analytics agent is on a deprecation path and won't be supported after August 31, 2024. You must migrate to the replacement 'Azure Monitor agent' prior to that date</p>	deployIfNotExists	3.0.0
Deploy the Linux Guest Configuration extension to enable Guest Configuration assignments on Linux VMs	<p>This policy deploys the Linux Guest Configuration extension to Linux virtual machines hosted in Azure that are supported by Guest Configuration. The Linux Guest Configuration extension is a prerequisite for all Linux Guest Configuration assignments and must be deployed to machines before using any Linux Guest Configuration policy definition. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	deployIfNotExists	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Deploy the Windows Guest Configuration extension to enable Guest Configuration assignments on Windows VMs</a>	<p>This policy deploys the Windows Guest Configuration extension to Windows virtual machines hosted in Azure that are supported by Guest Configuration. The Windows Guest Configuration extension is a prerequisite for all Windows Guest Configuration assignments and must be deployed to machines before using any Windows Guest Configuration policy definition. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	deployIfExists	1.2.0
<a href="#">Disk access resources should use private link</a>	<p>Azure Private Link lets you connect your virtual network to Azure services without a public IP address at the source or destination. The Private Link platform handles the connectivity between the consumer and services over the Azure backbone network. By mapping private endpoints to diskAccesses, data leakage risks are reduced. Learn more about private links at: <a href="https://aka.ms/disksprivatelinksdoc">https://aka.ms/disksprivatelinksdoc</a>.</p>	AuditIfExists, Disabled	1.0.0
<a href="#">Endpoint protection health issues should be resolved on your machines</a>	<p>Resolve endpoint protection health issues on your virtual machines to protect them from latest threats and vulnerabilities. Azure Security Center supported endpoint protection solutions are documented here - <a href="https://docs.microsoft.com/azure/security-center/security-center-services?tabs=features-windows#supported-endpoint-protection-solutions">https://docs.microsoft.com/azure/security-center/security-center-services?tabs=features-windows#supported-endpoint-protection-solutions</a>. Endpoint protection assessment is documented here - <a href="https://docs.microsoft.com/azure/security-center/security-center-endpoint-protection">https://docs.microsoft.com/azure/security-center/security-center-endpoint-protection</a>.</p>	AuditIfExists, Disabled	1.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Endpoint protection should be installed on your machines</a>	To protect your machines from threats and vulnerabilities, install a supported endpoint protection solution.	AuditIfNotExists, Disabled	1.0.0
<a href="#">Endpoint protection solution should be installed on virtual machine scale sets</a>	Audit the existence and health of an endpoint protection solution on your virtual machines scale sets, to protect them from threats and vulnerabilities.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Guest Configuration extension should be installed on your machines</a>	To ensure secure configurations of in-guest settings of your machine, install the Guest Configuration extension. In-guest settings that the extension monitors include the configuration of the operating system, application configuration or presence, and environment settings. Once installed, in-guest policies will be available such as 'Windows Exploit guard should be enabled'. Learn more at <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	1.0.2
<a href="#">Hotpatch should be enabled for Windows Server Azure Edition VMs</a>	Minimize reboots and install updates quickly with hotpatch. Learn more at <a href="https://docs.microsoft.com/azure/automanage/automate-hotpatch">https://docs.microsoft.com/azure/automanage/automate-hotpatch</a>	Audit, Deny, Disabled	1.0.0
<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	Protect your virtual machines from potential threats by restricting access to them with network security groups (NSG). Learn more about controlling traffic with NSGs at <a href="https://aka.ms/nsg-doc">https://aka.ms/nsg-doc</a>	AuditIfNotExists, Disabled	3.0.0
<a href="#">IP Forwarding on your virtual machine should be disabled</a>	Enabling IP forwarding on a virtual machine's NIC allows the machine to receive traffic addressed to other destinations. IP forwarding is rarely required (e.g., when using the VM as a network virtual appliance), and therefore, this should be reviewed by the network security team.	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Linux machines should meet requirements for the Azure compute security baseline</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Machines are non-compliant if the machine is not configured correctly for one of the recommendations in the Azure compute security baseline.</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Linux machines should only have local accounts that are allowed</a>	<p>Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p> <p>Managing user accounts using Azure Active Directory is a best practice for management of identities. Reducing local machine accounts helps prevent the proliferation of identities managed outside a central system. Machines are non-compliant if local user accounts exist that are enabled and not listed in the policy parameter.</p>	AuditIfNotExists, Disabled	2.0.0
<a href="#">Linux virtual machine scale sets should have Azure Monitor Agent installed</a>	<p>Linux virtual machine scale sets should be monitored and secured through the deployed Azure Monitor Agent. The Azure Monitor Agent collects telemetry data from the guest OS. This policy will audit virtual machine scale sets with supported OS images in supported regions. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a>.</p>	AuditIfNotExists, Disabled	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Linux virtual machines should have Azure Monitor Agent installed</a>	Linux virtual machines should be monitored and secured through the deployed Azure Monitor Agent. The Azure Monitor Agent collects telemetry data from the guest OS. This policy will audit virtual machines with supported OS images in supported regions. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	AuditIfNotExists, Disabled	2.0.0
<a href="#">Log Analytics agent should be installed on your Cloud Services (extended support) role instances</a>	Security Center collects data from your Cloud Services (extended support) role instances to monitor for security vulnerabilities and threats.	AuditIfNotExists, Disabled	2.0.0
<a href="#">Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring</a>	This policy audits any Windows/Linux virtual machines (VMs) if the Log Analytics agent is not installed which Security Center uses to monitor for security vulnerabilities and threats	AuditIfNotExists, Disabled	1.0.0
<a href="#">Log Analytics agent should be installed on your virtual machine scale sets for Azure Security Center monitoring</a>	Security Center collects data from your Azure virtual machines (VMs) to monitor for security vulnerabilities and threats.	AuditIfNotExists, Disabled	1.0.0
<a href="#">Log Analytics extension should be enabled in virtual machine scale sets for listed virtual machine images</a>	Reports virtual machine scale sets as non-compliant if the virtual machine image is not in the list defined and the extension is not installed.	AuditIfNotExists, Disabled	2.0.1

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Managed disks should be double encrypted with both platform-managed and customer-managed keys</a>	<p>High security sensitive customers who are concerned of the risk associated with any particular encryption algorithm, implementation, or key being compromised can opt for additional layer of encryption using a different encryption algorithm/mode at the infrastructure layer using platform managed encryption keys. The disk encryption sets are required to use double encryption.</p> <p>Learn more at <a href="https://aka.ms/disks-doubleEncryption">https://aka.ms/disks-doubleEncryption</a>.</p>	Audit, Deny, Disabled	1.0.0
<a href="#">Managed disks should disable public network access</a>	<p>Disabling public network access improves security by ensuring that a managed disk isn't exposed on the public internet. Creating private endpoints can limit exposure of managed disks.</p> <p>Learn more at: <a href="https://aka.ms/disksprivatenetworksdoc">https://aka.ms/disksprivatenetworksdoc</a>.</p>	Audit, Disabled	2.0.0
<a href="#">Managed disks should use a specific set of disk encryption sets for the customer-managed key encryption</a>	<p>Requiring a specific set of disk encryption sets to be used with managed disks give you control over the keys used for encryption at rest. You are able to select the allowed encrypted sets and all others are rejected when attached to a disk.</p> <p>Learn more at <a href="https://aka.ms/disks-cmk">https://aka.ms/disks-cmk</a>.</p>	Audit, Deny, Disabled	2.0.0
<a href="#">Management ports of virtual machines should be protected with just-in-time network access control</a>	<p>Possible network Just In Time (JIT) access will be monitored by Azure Security Center as recommendations</p>	AuditIfExists, Disabled	3.0.0
<a href="#">Management ports should be closed on your virtual machines</a>	<p>Open remote management ports are exposing your VM to a high level of risk from Internet-based attacks. These attacks attempt to brute force credentials to gain admin access to the machine.</p>	AuditIfExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Microsoft Antimalware for Azure should be configured to automatically update protection signatures</a>	This policy audits any Windows virtual machine not configured with automatic update of Microsoft Antimalware protection signatures.	AuditIfNotExists, Disabled	1.0.0
<a href="#">Microsoft IaaSAntimalware extension should be deployed on Windows servers</a>	This policy audits any Windows server VM without Microsoft IaaSAntimalware extension deployed.	AuditIfNotExists, Disabled	1.1.0
<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	Servers without an installed Endpoint Protection agent will be monitored by Azure Security Center as recommendations	AuditIfNotExists, Disabled	3.0.0
<a href="#">Non-internet-facing virtual machines should be protected with network security groups</a>	Protect your non-internet-facing virtual machines from potential threats by restricting access with network security groups (NSG). Learn more about controlling traffic with NSGs at <a href="https://aka.ms/nsg-doc">https://aka.ms/nsg-doc</a>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Only approved VM extensions should be installed</a>	This policy governs the virtual machine extensions that are not approved.	Audit, Deny, Disabled	1.0.0
<a href="#">OS and data disks should be encrypted with a customer-managed key</a>	Use customer-managed keys to manage the encryption at rest of the contents of your managed disks. By default, the data is encrypted at rest with platform-managed keys, but customer-managed keys are commonly required to meet regulatory compliance standards. Customer-managed keys enable the data to be encrypted with an Azure Key Vault key created and owned by you. You have full control and responsibility for the key lifecycle, including rotation and management. Learn more at <a href="https://aka.ms/disks-cmk">https://aka.ms/disks-cmk</a> .	Audit, Deny, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
Require automatic OS image patching on Virtual Machine Scale Sets	This policy enforces enabling automatic OS image patching on Virtual Machine Scale Sets to always keep Virtual Machines secure by safely applying latest security patches every month.	deny	1.0.0
Resource logs in Virtual Machine Scale Sets should be enabled	It is recommended to enable Logs so that activity trail can be recreated when investigations are required in the event of an incident or a compromise.	AuditIfNotExists, Disabled	2.1.0
SQL servers on machines should have vulnerability findings resolved	SQL vulnerability assessment scans your database for security vulnerabilities, and exposes any deviations from best practices such as misconfigurations, excessive permissions, and unprotected sensitive data. Resolving the vulnerabilities found can greatly improve your database security posture.	AuditIfNotExists, Disabled	1.0.0
System updates on virtual machine scale sets should be installed	Audit whether there are any missing system security updates and critical updates that should be installed to ensure that your Windows and Linux virtual machine scale sets are secure.	AuditIfNotExists, Disabled	3.0.0
System updates should be installed on your machines	Missing security system updates on your servers will be monitored by Azure Security Center as recommendations	AuditIfNotExists, Disabled	4.0.0
The Log Analytics extension should be installed on Virtual Machine Scale Sets	This policy audits any Windows/Linux Virtual Machine Scale Sets if the Log Analytics extension is not installed.	AuditIfNotExists, Disabled	1.0.1

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Virtual machines and virtual machine scale sets should have encryption at host enabled</a>	<p>Use encryption at host to get end-to-end encryption for your virtual machine and virtual machine scale set data. Encryption at host enables encryption at rest for your temporary disk and OS/data disk caches. Temporary and ephemeral OS disks are encrypted with platform-managed keys when encryption at host is enabled. OS/data disk caches are encrypted at rest with either customer-managed or platform-managed key, depending on the encryption type selected on the disk. Learn more at <a href="https://aka.ms/vm-hbe">https://aka.ms/vm-hbe</a>.</p>	Audit, Deny, Disabled	1.0.0
<a href="#">Virtual machines should be connected to a specified workspace</a>	<p>Reports virtual machines as non-compliant if they aren't logging to the Log Analytics workspace specified in the policy/initiative assignment.</p>	AuditIfNotExists, Disabled	1.1.0
<a href="#">Virtual machines should be migrated to new Azure Resource Manager resources</a>	<p>Use new Azure Resource Manager for your virtual machines to provide security enhancements such as: stronger access control (RBAC), better auditing, Azure Resource Manager based deployment and governance, access to managed identities, access to key vault for secrets, Azure AD-based authentication and support for tags and resource groups for easier security management</p>	Audit, Deny, Disabled	1.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources</a>	<p>By default, a virtual machine's OS and data disks are encrypted-at-rest using platform-managed keys. Temp disks, data caches and data flowing between compute and storage aren't encrypted. Disregard this recommendation if: 1. using encryption-at-host, or 2. server-side encryption on Managed Disks meets your security requirements.</p> <p>Learn more in: Server-side encryption of Azure Disk Storage:  <a href="https://aka.ms/disksse">https://aka.ms/disksse</a>,          Different disk encryption offerings:  <a href="https://aka.ms/diskencryptioncomparison">https://aka.ms/diskencryptioncomparison</a></p>	AuditIfNotExists, Disabled	2.0.3
<a href="#">Virtual machines should have the Log Analytics extension installed</a>	This policy audits any Windows/Linux virtual machines if the Log Analytics extension is not installed.	AuditIfNotExists, Disabled	1.0.1
<a href="#">Virtual machines' Guest Configuration extension should be deployed with system-assigned managed identity</a>	The Guest Configuration extension requires a system assigned managed identity. Azure virtual machines in the scope of this policy will be non-compliant when they have the Guest Configuration extension installed but do not have a system assigned managed identity. Learn more at <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>	AuditIfNotExists, Disabled	1.0.1
<a href="#">Vulnerabilities in container security configurations should be remediated</a>	Audit vulnerabilities in security configuration on machines with Docker installed and display as recommendations in Azure Security Center.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	Servers which do not satisfy the configured baseline will be monitored by Azure Security Center as recommendations	AuditIfNotExists, Disabled	3.0.0
<a href="#">Vulnerabilities in security configuration on your virtual machine scale sets should be remediated</a>	Audit the OS vulnerabilities on your virtual machine scale sets to protect them from attacks.	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows Defender Exploit Guard should be enabled on your machines</a>	Windows Defender Exploit Guard uses the Azure Policy Guest Configuration agent. Exploit Guard has four components that are designed to lock down devices against a wide variety of attack vectors and block behaviors commonly used in malware attacks while enabling enterprises to balance their security risk and productivity requirements (Windows only).	AuditIfNotExists, Disabled	2.0.0
<a href="#">Windows machines should configure Windows Defender to update protection signatures within one day</a>	To provide adequate protection against newly released malware, Windows Defender protection signatures need to be updated regularly to account for newly released malware. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	1.0.0
<a href="#">Windows machines should enable Windows Defender Real-time protection</a>	Windows machines should enable the Real-time protection in the Windows Defender to provide adequate protection against newly released malware. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	1.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Administrative Templates - Control Panel'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'Administrative Templates - Control Panel' for input personalization and prevention of enabling lock screens. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope.</p> <p>For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Administrative Templates - MSS (Legacy)'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'Administrative Templates - MSS (Legacy)' for automatic logon, screen saver, network behavior, safe DLL, and event log.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Administrative Templates - Network'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'Administrative Templates - Network' for guest logons, simultaneous connections, network bridge, ICS, and multicast name resolution. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope.</p> <p>For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Administrative Templates - System'</a>	Windows machines should have the specified Group Policy settings in the category 'Administrative Templates - System' for settings that control the administrative experience and Remote Assistance. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Accounts'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Accounts' for limiting local account use of blank passwords and guest account status. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Audit'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Audit' for forcing audit policy subcategory and shutting down if unable to log security audits. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Security Options - Devices'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Devices' for undocking without logging on, installing print drivers, and formatting/ejecting media. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Interactive Logon'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Interactive Logon' for displaying last user name and requiring ctrl-alt-del. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Microsoft Network Client'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Microsoft Network Client' for Microsoft network client/server and SMB v1. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Microsoft Network Server'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Microsoft Network Server' for disabling SMB v1 server. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Security Options - Network Access'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Network Access' for including access for anonymous users, local accounts, and remote access to the registry. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Network Security'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Network Security' for including Local System behavior, PKU2U, LAN Manager, LDAP client, and NTLM SSP. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - Recovery console'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Recovery console' for allowing floppy copy and access to all drives and folders. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Security Options - Shutdown'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - Shutdown' for allowing shutdown without logon and clearing the virtual memory pagefile. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - System objects'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - System objects' for case insensitivity for non- Windows subsystems and permissions of internal system objects. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Options - System settings'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - System settings' for certificate rules on executables for SRP and optional subsystems. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Security Options - User Account Control'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Options - User Account Control' for mode for admins, behavior of elevation prompt, and virtualizing file and registry write failures. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Security Settings - Account Policies'</a>	Windows machines should have the specified Group Policy settings in the category 'Security Settings - Account Policies' for password history, age, length, complexity, and storing passwords using reversible encryption. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Account Logon'</a>	Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Account Logon' for auditing credential validation and other account logon events. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Account Management'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Account Management' for auditing application, security, and user group management, and other management events. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Detailed Tracking'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Detailed Tracking' for auditing DPAPI, process creation/termination, RPC events, and PNP activity. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Logon-Logoff'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Logon-Logoff' for auditing IPSec, network policy, claims, account lockout, group membership, and logon/logoff events. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Object Access'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Object Access' for auditing file, registry, SAM, storage, filtering, kernel, and other system types.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Policy Change'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Policy Change' for auditing changes to system audit policies. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope.</p> <p>For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'System Audit Policies - Privilege Use'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - Privilege Use' for auditing nonsensitive and other privilege use. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'System Audit Policies - System'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'System Audit Policies - System' for auditing IPsec driver, system integrity, system extension, state change, and other system events.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'User Rights Assignment'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'User Rights Assignment' for allowing log on locally, RDP, access from the network, and many other user activities.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements for 'Windows Components'</a>	<p>Windows machines should have the specified Group Policy settings in the category 'Windows Components' for basic authentication, unencrypted traffic, Microsoft accounts, telemetry, Cortana, and other Windows behaviors.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should meet requirements for 'Windows Firewall Properties'</a>	Windows machines should have the specified Group Policy settings in the category 'Windows Firewall Properties' for firewall state, connections, rule management, and notifications. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows machines should meet requirements of the Azure compute security baseline</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . Machines are non-compliant if the machine is not configured correctly for one of the recommendations in the Azure compute security baseline.	AuditIfNotExists, Disabled	2.0.0
<a href="#">Windows machines should only have local accounts that are allowed</a>	Requires that prerequisites are deployed to the policy assignment scope. For details, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a> . This definition is not supported on Windows Server 2012 or 2012 R2. Managing user accounts using Azure Active Directory is a best practice for management of identities. Reducing local machine accounts helps prevent the proliferation of identities managed outside a central system. Machines are non-compliant if local user accounts exist that are enabled and not listed in the policy parameter.	AuditIfNotExists, Disabled	2.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows machines should schedule Windows Defender to perform a scheduled scan every day</a>	<p>Windows machines should schedule Windows Defender to perform a scheduled scan every day to ensure that malware is quickly identified to minimize the effect this may have to the environment.</p> <p>This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	1.0.0
<a href="#">Windows machines should use the default NTP server</a>	<p>Setup the 'time.windows.com' as the default NTP Server for all Windows machines to ensure logs across all systems have system clocks that are all in sync. This policy requires that the Guest Configuration prerequisites have been deployed to the policy assignment scope. For more information on Guest Configuration, visit <a href="https://aka.ms/gcpol">https://aka.ms/gcpol</a>.</p>	AuditIfNotExists, Disabled	1.0.0
<a href="#">Windows virtual machine scale sets should have Azure Monitor Agent installed</a>	<p>Windows virtual machine scale sets should be monitored and secured through the deployed Azure Monitor Agent. The Azure Monitor Agent collects telemetry data from the guest OS. Virtual machine scale sets with supported OS and in supported regions are monitored for Azure Monitor Agent deployment.</p> <p>Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a>.</p>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Windows virtual machines should have Azure Monitor Agent installed</a>	Windows virtual machines should be monitored and secured through the deployed Azure Monitor Agent. The Azure Monitor Agent collects telemetry data from the guest OS. Windows virtual machines with supported OS and in supported regions are monitored for Azure Monitor Agent deployment. Learn more: <a href="https://aka.ms/AMAOerview">https://aka.ms/AMAOerview</a> .	AuditIfNotExists, Disabled	3.0.0
<a href="#">Windows web servers should be configured to use secure communication protocols</a>	To protect the privacy of information communicated over the Internet, your web servers should use the latest version of the industry-standard cryptographic protocol, Transport Layer Security (TLS). TLS secures communications over a network by using security certificates to encrypt a connection between machines.	AuditIfNotExists, Disabled	4.0.0

## Microsoft.VirtualMachineImages

NAME (AZURE PORTAL)	DESCRIPTION	EFFECT(S)	VERSION (GITHUB)
<a href="#">VM Image Builder templates should use private link</a>	Azure Private Link lets you connect your virtual network to Azure services without a public IP address at the source or destination. The Private Link platform handles the connectivity between the consumer and services over the Azure backbone network. By mapping private endpoints to your VM Image Builder building resources, data leakage risks are reduced. Learn more about private links at: <a href="https://docs.microsoft.com/azure/virtual-machines/linux/image-builder-networking#deploy-using-an-existing-vnet">https://docs.microsoft.com/azure/virtual-machines/linux/image-builder-networking#deploy-using-an-existing-vnet</a> .	Audit, Disabled, Deny	1.1.0

# Microsoft.ClassicCompute

NAME (AZURE PORTAL)	DESCRIPTION	EFFECT(S)	VERSION (GITHUB)
<a href="#">A vulnerability assessment solution should be enabled on your virtual machines</a>	Audits virtual machines to detect whether they are running a supported vulnerability assessment solution. A core component of every cyber risk and security program is the identification and analysis of vulnerabilities. Azure Security Center's standard pricing tier includes vulnerability scanning for your virtual machines at no extra cost. Additionally, Security Center can automatically deploy this tool for you.	AuditIfNotExists, Disabled	<a href="#">3.0.0</a>
<a href="#">Adaptive application controls for defining safe applications should be enabled on your machines</a>	Enable application controls to define the list of known-safe applications running on your machines, and alert you when other applications run. This helps harden your machines against malware. To simplify the process of configuring and maintaining your rules, Security Center uses machine learning to analyze the applications running on each machine and suggest the list of known-safe applications.	AuditIfNotExists, Disabled	<a href="#">3.0.0</a>
<a href="#">All network ports should be restricted on network security groups associated to your virtual machine</a>	Azure Security Center has identified some of your network security groups' inbound rules to be too permissive. Inbound rules should not allow access from 'Any' or 'Internet' ranges. This can potentially enable attackers to target your resources.	AuditIfNotExists, Disabled	<a href="#">3.0.0</a>

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Allowlist rules in your adaptive application control policy should be updated</a>	Monitor for changes in behavior on groups of machines configured for auditing by Azure Security Center's adaptive application controls. Security Center uses machine learning to analyze the running processes on your machines and suggest a list of known-safe applications. These are presented as recommended apps to allow in adaptive application control policies.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Audit virtual machines without disaster recovery configured</a>	Audit virtual machines which do not have disaster recovery configured. To learn more about disaster recovery, visit <a href="https://aka.ms/asr-doc">https://aka.ms/asr-doc</a> .	auditIfNotExists	1.0.0
<a href="#">Endpoint protection health issues should be resolved on your machines</a>	Resolve endpoint protection health issues on your virtual machines to protect them from latest threats and vulnerabilities. Azure Security Center supported endpoint protection solutions are documented here - <a href="https://docs.microsoft.com/azure/security-center/security-center-services?tabs=features-windows#supported-endpoint-protection-solutions">https://docs.microsoft.com/azure/security-center/security-center-services?tabs=features-windows#supported-endpoint-protection-solutions</a> . Endpoint protection assessment is documented here - <a href="https://docs.microsoft.com/azure/security-center/security-center-endpoint-protection">https://docs.microsoft.com/azure/security-center/security-center-endpoint-protection</a> .	AuditIfNotExists, Disabled	1.0.0
<a href="#">Endpoint protection should be installed on your machines</a>	To protect your machines from threats and vulnerabilities, install a supported endpoint protection solution.	AuditIfNotExists, Disabled	1.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">Internet-facing virtual machines should be protected with network security groups</a>	Protect your virtual machines from potential threats by restricting access to them with network security groups (NSG). Learn more about controlling traffic with NSGs at <a href="https://aka.ms/nsg-doc">https://aka.ms/nsg-doc</a>	AuditIfNotExists, Disabled	3.0.0
<a href="#">IP Forwarding on your virtual machine should be disabled</a>	Enabling IP forwarding on a virtual machine's NIC allows the machine to receive traffic addressed to other destinations. IP forwarding is rarely required (e.g., when using the VM as a network virtual appliance), and therefore, this should be reviewed by the network security team.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Log Analytics agent should be installed on your virtual machine for Azure Security Center monitoring</a>	This policy audits any Windows/Linux virtual machines (VMs) if the Log Analytics agent is not installed which Security Center uses to monitor for security vulnerabilities and threats	AuditIfNotExists, Disabled	1.0.0
<a href="#">Management ports should be closed on your virtual machines</a>	Open remote management ports are exposing your VM to a high level of risk from Internet-based attacks. These attacks attempt to brute force credentials to gain admin access to the machine.	AuditIfNotExists, Disabled	3.0.0
<a href="#">Monitor missing Endpoint Protection in Azure Security Center</a>	Servers without an installed Endpoint Protection agent will be monitored by Azure Security Center as recommendations	AuditIfNotExists, Disabled	3.0.0
<a href="#">Non-internet-facing virtual machines should be protected with network security groups</a>	Protect your non-internet-facing virtual machines from potential threats by restricting access with network security groups (NSG). Learn more about controlling traffic with NSGs at <a href="https://aka.ms/nsg-doc">https://aka.ms/nsg-doc</a>	AuditIfNotExists, Disabled	3.0.0

NAME	DESCRIPTION	EFFECT(S)	VERSION
<a href="#">System updates should be installed on your machines</a>	Missing security system updates on your servers will be monitored by Azure Security Center as recommendations	AuditIfExists, Disabled	4.0.0
<a href="#">Virtual machines should be migrated to new Azure Resource Manager resources</a>	Use new Azure Resource Manager for your virtual machines to provide security enhancements such as: stronger access control (RBAC), better auditing, Azure Resource Manager based deployment and governance, access to managed identities, access to key vault for secrets, Azure AD-based authentication and support for tags and resource groups for easier security management	Audit, Deny, Disabled	1.0.0
<a href="#">Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources</a>	By default, a virtual machine's OS and data disks are encrypted-at-rest using platform-managed keys. Temp disks, data caches and data flowing between compute and storage aren't encrypted. Disregard this recommendation if: 1. using encryption-at-host, or 2. server-side encryption on Managed Disks meets your security requirements. Learn more in: Server-side encryption of Azure Disk Storage: <a href="https://aka.ms/disksse">https://aka.ms/disksse</a> , Different disk encryption offerings: <a href="https://aka.ms/diskencryptioncomparison">https://aka.ms/diskencryptioncomparison</a>	AuditIfExists, Disabled	2.0.3
<a href="#">Vulnerabilities in container security configurations should be remediated</a>	Audit vulnerabilities in security configuration on machines with Docker installed and display as recommendations in Azure Security Center.	AuditIfExists, Disabled	3.0.0
<a href="#">Vulnerabilities in security configuration on your machines should be remediated</a>	Servers which do not satisfy the configured baseline will be monitored by Azure Security Center as recommendations	AuditIfExists, Disabled	3.0.0

## Next steps

- See the built-ins on the [Azure Policy GitHub repo](#).
- Review the [Azure Policy definition structure](#).
- Review [Understanding policy effects](#).

# Apply policies to Linux VMs with Azure Resource Manager

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

By using policies, an organization can enforce various conventions and rules throughout the enterprise. Enforcement of the desired behavior can help mitigate risk while contributing to the success of the organization. In this article, we describe how you can use Azure Resource Manager policies to define the desired behavior for your organization's Virtual Machines.

For an introduction to policies, see [What is Azure Policy?](#).

## Permitted Virtual Machines

To ensure that virtual machines for your organization are compatible with an application, you can restrict the permitted operating systems. In the following policy example, you allow only Ubuntu 14.04.2-LTS Virtual Machines to be created.

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "in": [
          "Microsoft.Compute/virtualMachines",
          "Microsoft.Compute/VirtualMachineScaleSets"
        ]
      },
      {
        "not": {
          "allOf": [
            {
              "field": "Microsoft.Compute/imagePublisher",
              "in": [
                "Canonical"
              ]
            },
            {
              "field": "Microsoft.Compute/imageOffer",
              "in": [
                "UbuntuServer"
              ]
            },
            {
              "field": "Microsoft.Compute/imageSku",
              "in": [
                "14.04.2-LTS"
              ]
            },
            {
              "field": "Microsoft.Compute/imageVersion",
              "in": [
                "latest"
              ]
            }
          ]
        }
      ],
      "then": {
        "effect": "deny"
      }
    }
  }
}
```

Use a wild card to modify the preceding policy to allow any Ubuntu LTS image:

```
{
  "field": "Microsoft.Compute/virtualMachines/imageSku",
  "like": "*LTS"
}
```

For information about policy fields, see [Policy aliases](#).

## Managed disks

To require the use of managed disks, use the following policy:

```
{
  "if": {
    "anyOf": [
      {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Compute/virtualMachines"
          },
          {
            "field": "Microsoft.Compute/virtualMachines/osDisk.uri",
            "exists": true
          }
        ]
      },
      {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Compute/VirtualMachineScaleSets"
          },
          {
            "anyOf": [
              {
                "field": "Microsoft.Compute/VirtualMachineScaleSets/osDisk.vhdContainers",
                "exists": true
              },
              {
                "field": "Microsoft.Compute/VirtualMachineScaleSets/osdisk.imageUrl",
                "exists": true
              }
            ]
          }
        ]
      }
    ],
    "then": {
      "effect": "deny"
    }
  }
}
```

## Images for Virtual Machines

For security reasons, you can require that only approved custom images are deployed in your environment. You can specify either the resource group that contains the approved images, or the specific approved images.

The following example requires images from an approved resource group:

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "in": [
          "Microsoft.Compute/virtualMachines",
          "Microsoft.Compute/VirtualMachineScaleSets"
        ]
      },
      {
        "not": {
          "field": "Microsoft.Compute/imageId",
          "contains": "resourceGroups/CustomImage"
        }
      }
    ],
    "then": {
      "effect": "deny"
    }
  }
}
```

The following example specifies the approved image IDs:

```
{
  "field": "Microsoft.Compute/imageId",
  "in": ["{imageId1}", "{imageId2}"]
}
```

## Virtual Machine extensions

You may want to forbid usage of certain types of extensions. For example, an extension may not be compatible with certain custom virtual machine images. The following example shows how to block a specific extension. It uses publisher and type to determine which extension to block.

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Compute/virtualMachines/extensions"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/publisher",
        "equals": "Microsoft.Compute"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/type",
        "equals": "{extension-type}"
      }
    ],
    "then": {
      "effect": "deny"
    }
  }
}
```

## Next steps

- After defining a policy rule (as shown in the preceding examples), you need to create the policy definition and assign it to a scope. The scope can be a subscription, resource group, or resource. To assign policies, see [Use Azure portal to assign and manage resource policies](#), [Use PowerShell to assign policies](#), or [Use Azure CLI to assign policies](#).
- For an introduction to resource policies, see [What is Azure Policy?](#).
- For guidance on how enterprises can use Resource Manager to effectively manage subscriptions, see [Azure enterprise scaffold - prescriptive subscription governance](#).

# Apply policies to Windows VMs with Azure Resource Manager

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

By using policies, an organization can enforce various conventions and rules throughout the enterprise. Enforcement of the desired behavior can help mitigate risk while contributing to the success of the organization. In this article, we describe how you can use Azure Resource Manager policies to define the desired behavior for your organization's Virtual Machines.

For an introduction to policies, see [What is Azure Policy?](#).

## Permitted Virtual Machines

To ensure that virtual machines for your organization are compatible with an application, you can restrict the permitted operating systems. In the following policy example, you allow only Windows Server 2012 R2 Datacenter Virtual Machines to be created:

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "in": [
          "Microsoft.Compute/virtualMachines",
          "Microsoft.Compute/VirtualMachineScaleSets"
        ]
      },
      {
        "not": {
          "allOf": [
            {
              "field": "Microsoft.Compute/imagePublisher",
              "in": [
                "MicrosoftWindowsServer"
              ]
            },
            {
              "field": "Microsoft.Compute/imageOffer",
              "in": [
                "WindowsServer"
              ]
            },
            {
              "field": "Microsoft.Compute/imageSku",
              "in": [
                "2012-R2-Datacenter"
              ]
            },
            {
              "field": "Microsoft.Compute/imageVersion",
              "in": [
                "latest"
              ]
            }
          ]
        }
      ],
      "then": {
        "effect": "deny"
      }
    }
  }
}
```

Use a wild card to modify the preceding policy to allow any Windows Server Datacenter image:

```
{
  "field": "Microsoft.Compute/imageSku",
  "like": "*Datacenter"
}
```

Use anyOf to modify the preceding policy to allow any Windows Server 2012 R2 Datacenter or higher image:

```
{
  "anyOf": [
    {
      "field": "Microsoft.Compute/imageSku",
      "like": "2012-R2-Datacenter*"
    },
    {
      "field": "Microsoft.Compute/imageSku",
      "like": "2016-Datacenter*"
    }
  ]
}
```

For information about policy fields, see [Policy aliases](#).

## Managed disks

To require the use of managed disks, use the following policy:

```
{
  "if": {
    "anyOf": [
      {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Compute/virtualMachines"
          },
          {
            "field": "Microsoft.Compute/virtualMachines/osDisk.uri",
            "exists": true
          }
        ]
      },
      {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Compute/VirtualMachineScaleSets"
          },
          {
            "anyOf": [
              {
                "field": "Microsoft.Compute/VirtualMachineScaleSets/osDisk.vhdContainers",
                "exists": true
              },
              {
                "field": "Microsoft.Compute/VirtualMachineScaleSets/osdisk.imageUrl",
                "exists": true
              }
            ]
          }
        ]
      }
    ],
    "then": {
      "effect": "deny"
    }
  }
}
```

## Images for Virtual Machines

For security reasons, you can require that only approved custom images are deployed in your environment. You can specify either the resource group that contains the approved images, or the specific approved images.

The following example requires images from an approved resource group:

```
{  
  "if": {  
    "allOf": [  
      {  
        "field": "type",  
        "in": [  
          "Microsoft.Compute/virtualMachines",  
          "Microsoft.Compute/VirtualMachineScaleSets"  
        ]  
      },  
      {  
        "not": {  
          "field": "Microsoft.Compute/imageId",  
          "contains": "resourceGroups/CustomImage"  
        }  
      }  
    ],  
    "then": {  
      "effect": "deny"  
    }  
  }  
}
```

The following example specifies the approved image IDs:

```
{  
  "field": "Microsoft.Compute/imageId",  
  "in": ["{imageId1}","{imageId2}"]  
}
```

## Virtual Machine extensions

You may want to forbid usage of certain types of extensions. For example, an extension may not be compatible with certain custom virtual machine images. The following example shows how to block a specific extension. It uses publisher and type to determine which extension to block.

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "equals": "Microsoft.Compute/virtualMachines/extensions"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/publisher",
        "equals": "Microsoft.Compute"
      },
      {
        "field": "Microsoft.Compute/virtualMachines/extensions/type",
        "equals": "{extension-type}"
      }
    ]
  },
  "then": {
    "effect": "deny"
  }
}
```

## Azure Hybrid Use Benefit

When you have an on-premises license, you can save the license fee on your virtual machines. When you don't have the license, you should forbid the option. The following policy forbids usage of Azure Hybrid Use Benefit (AHUB):

```
{
  "if": {
    "allOf": [
      {
        "field": "type",
        "in": [ "Microsoft.Compute/virtualMachines", "Microsoft.Compute/VirtualMachineScaleSets" ]
      },
      {
        "field": "Microsoft.Compute/licenseType",
        "exists": true
      }
    ]
  },
  "then": {
    "effect": "deny"
  }
}
```

## Next steps

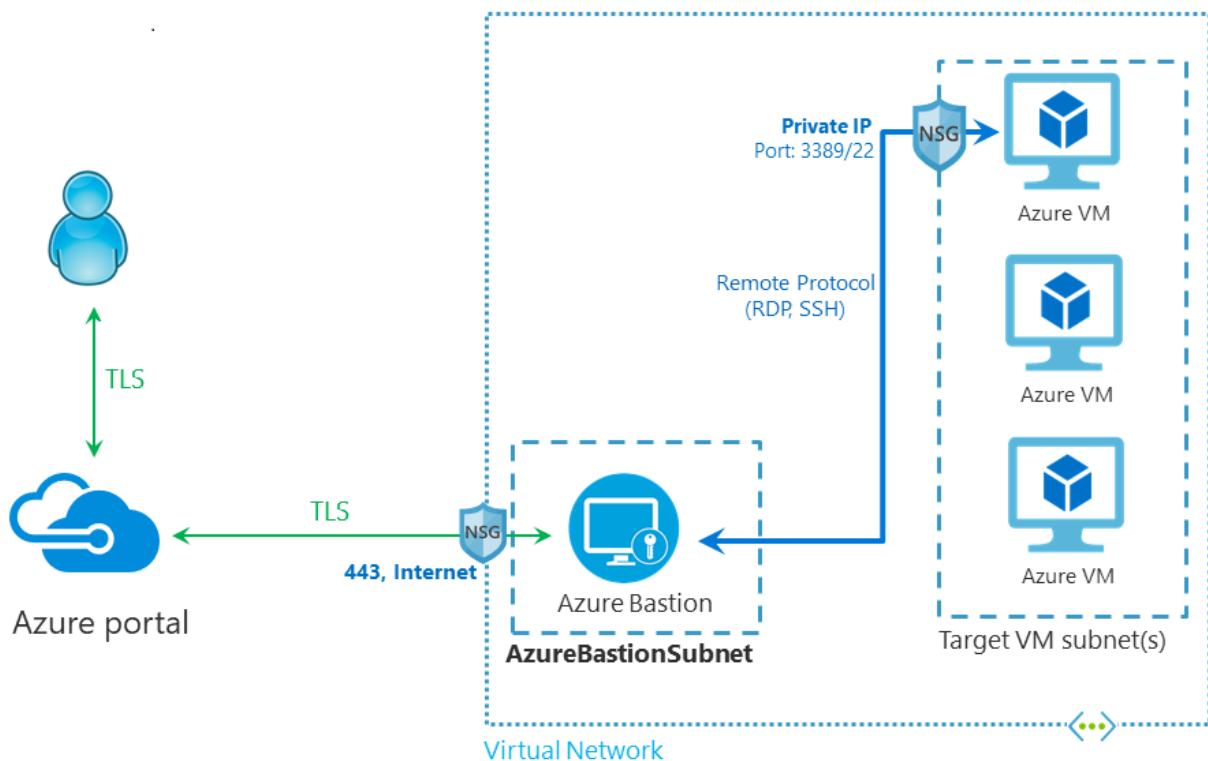
- After defining a policy rule (as shown in the preceding examples), you need to create the policy definition and assign it to a scope. The scope can be a subscription, resource group, or resource. To assign policies, see [Use Azure portal to assign and manage resource policies](#), [Use PowerShell to assign policies](#), or [Use Azure CLI to assign policies](#).
- For an introduction to resource policies, see [What is Azure Policy?](#).
- For guidance on how enterprises can use Resource Manager to effectively manage subscriptions, see [Azure enterprise scaffold - prescriptive subscription governance](#).

# What is Azure Bastion?

9/21/2022 • 4 minutes to read • [Edit Online](#)

Azure Bastion is a service you deploy that lets you connect to a virtual machine using your browser and the Azure portal, or via the native SSH or RDP client already installed on your local computer. The Azure Bastion service is a fully platform-managed PaaS service that you provision inside your virtual network. It provides secure and seamless RDP/SSH connectivity to your virtual machines directly from the Azure portal over TLS. When you connect via Azure Bastion, your virtual machines don't need a public IP address, agent, or special client software.

Bastion provides secure RDP and SSH connectivity to all of the VMs in the virtual network in which it is provisioned. Using Azure Bastion protects your virtual machines from exposing RDP/SSH ports to the outside world, while still providing secure access using RDP/SSH.



## Key benefits

BENEFIT	DESCRIPTION
RDP and SSH through the Azure portal	You can get to the RDP and SSH session directly in the Azure portal using a single-click seamless experience.
Remote Session over TLS and firewall traversal for RDP/SSH	Azure Bastion uses an HTML5 based web client that is automatically streamed to your local device. Your RDP/SSH session is over TLS on port 443. This enables the traffic to traverse firewalls more securely.

BENEFIT	DESCRIPTION
No Public IP address required on the Azure VM	Azure Bastion opens the RDP/SSH connection to your Azure VM by using the private IP address on your VM. You don't need a public IP address on your virtual machine.
No hassle of managing Network Security Groups (NSGs)	You don't need to apply any NSGs to the Azure Bastion subnet. Because Azure Bastion connects to your virtual machines over private IP, you can configure your NSGs to allow RDP/SSH from Azure Bastion only. This removes the hassle of managing NSGs each time you need to securely connect to your virtual machines. For more information about NSGs, see <a href="#">Network Security Groups</a> .
No need to manage a separate bastion host on a VM	Azure Bastion is a fully managed platform PaaS service from Azure that is hardened internally to provide you secure RDP/SSH connectivity.
Protection against port scanning	Your VMs are protected against port scanning by rogue and malicious users because you don't need to expose the VMs to the internet.
Hardening in one place only	Azure Bastion sits at the perimeter of your virtual network, so you don't need to worry about hardening each of the VMs in your virtual network.
Protection against zero-day exploits	The Azure platform protects against zero-day exploits by keeping the Azure Bastion hardened and always up to date for you.

## SKUs

Azure Bastion has two available SKUs, Basic and Standard. For more information, including how to upgrade a SKU, see the [Configuration settings](#) article.

The following table shows features and corresponding SKUs.

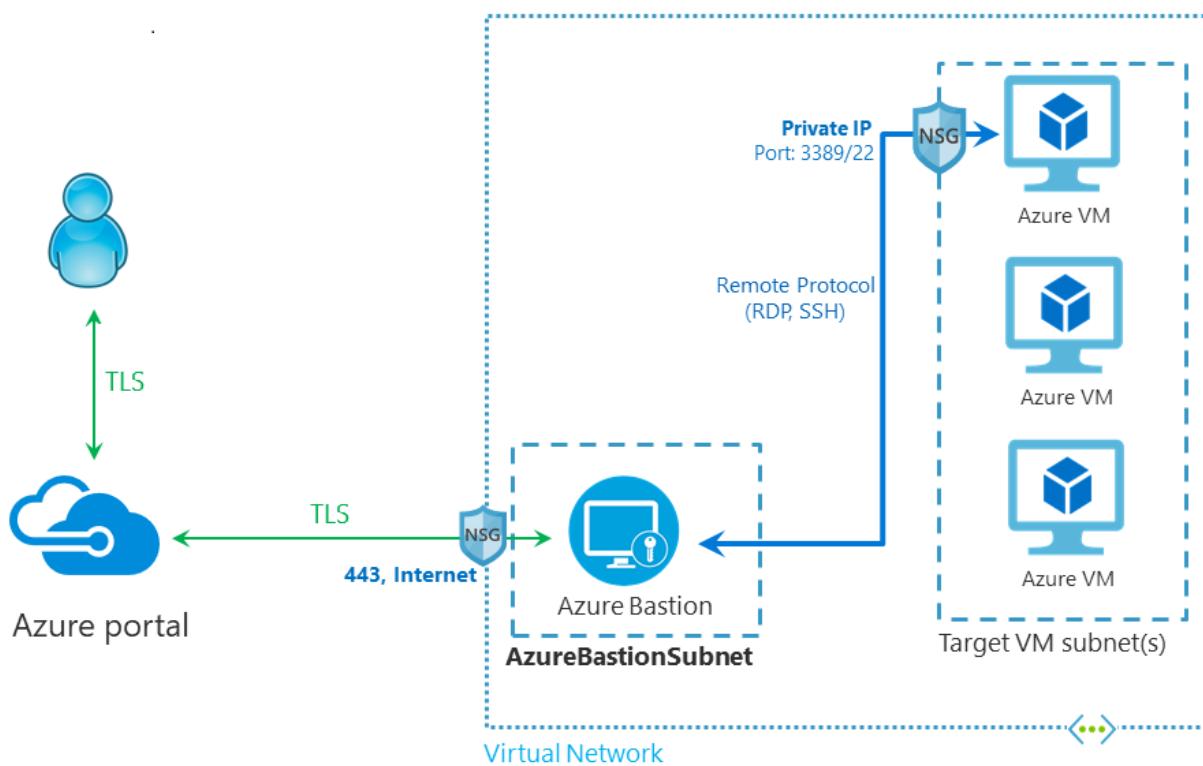
FEATURE	BASIC SKU	STANDARD SKU
Connect to target VMs in peered virtual networks	Yes	Yes
Access Linux VM Private Keys in Azure Key Vault (AKV)	Yes	Yes
Connect to Linux VM using SSH	Yes	Yes
Connect to Windows VM using RDP	Yes	Yes
VM audio output	Yes	Yes
Host scaling	Not available	Yes
Specify custom inbound port	Not available	Yes
Connect to Linux VM using RDP	Not available	Yes

FEATURE	BASIC SKU	STANDARD SKU
Connect to Windows VM using SSH	Not available	Yes
Upload or download files	Not available	Yes
Disable copy/paste (web-based clients)	Not available	Yes

## Architecture

Azure Bastion is deployed to a virtual network and supports virtual network peering. Specifically, Azure Bastion manages RDP/SSH connectivity to VMs created in the local or peered virtual networks.

RDP and SSH are some of the fundamental means through which you can connect to your workloads running in Azure. Exposing RDP/SSH ports over the Internet isn't desired and is seen as a significant threat surface. This is often due to protocol vulnerabilities. To contain this threat surface, you can deploy bastion hosts (also known as jump-servers) at the public side of your perimeter network. Bastion host servers are designed and configured to withstand attacks. Bastion servers also provide RDP and SSH connectivity to the workloads sitting behind the bastion, as well as further inside the network.



This figure shows the architecture of an Azure Bastion deployment. In this diagram:

- The Bastion host is deployed in the virtual network that contains the AzureBastionSubnet subnet that has a minimum /26 prefix.
- The user connects to the Azure portal using any HTML5 browser.
- The user selects the virtual machine to connect to.
- With a single click, the RDP/SSH session opens in the browser.
- No public IP is required on the Azure VM.

## Host scaling

Azure Bastion supports manual host scaling. You can configure the number of host instances (scale units) in order to manage the number of concurrent RDP/SSH connections that Azure Bastion can support. Increasing the number of host instances lets Azure Bastion manage more concurrent sessions. Decreasing the number of instances decreases the number of concurrent supported sessions. Azure Bastion supports up to 50 host instances. This feature is available for the Azure Bastion Standard SKU only.

For more information, see the [Configuration settings](#) article.

## Pricing

Azure Bastion pricing involves a combination of hourly pricing based on SKU, scale units, and data transfer rates. Pricing information can be found on the [Pricing](#) page.

## What's new?

Subscribe to the RSS feed and view the latest Azure Bastion feature updates on the [Azure Updates](#) page.

## Bastion FAQ

For frequently asked questions, see the Bastion [FAQ](#).

## Next steps

- [Quickstart: Deploy Bastion using default settings](#).
- [Tutorial: Deploy Bastion using specified settings](#).
- [Learn module: Introduction to Azure Bastion](#).
- Learn about some of the other key [networking capabilities](#) of Azure.

# Tutorial: Deploy Bastion using specified settings

9/21/2022 • 9 minutes to read • [Edit Online](#)

This tutorial helps you deploy Azure Bastion from the Azure portal using your own specified manual settings. When you use manual settings, you can specify configuration values such as instance counts and the SKU at the time of deployment. After Bastion is deployed, you can connect (SSH/RDP) to virtual machines in the virtual network via Bastion using the private IP address of the VM. When you connect to a VM, it doesn't need a public IP address, client software, agent, or a special configuration.

In this tutorial, you deploy Bastion using the Standard SKU tier and adjust host scaling (instance count). After the deployment is complete, you connect to your VM via private IP address. If your VM has a public IP address that you don't need for anything else, you can remove it.

Azure Bastion is a PaaS service that's maintained for you, not a bastion host that you install on one of your VMs and maintain yourself. For more information about Azure Bastion, see [What is Azure Bastion?](#)

In this tutorial, you'll learn how to:

- Deploy Bastion to your VNet.
- Connect to a virtual machine.
- Remove the public IP address from a virtual machine.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Prerequisites

- A [virtual network](#). This will be the VNet to which you deploy Bastion.
- A virtual machine in the virtual network. This VM isn't a part of the Bastion configuration and doesn't become a bastion host. You connect to this VM later in this tutorial via Bastion. If you don't have a VM, create one using [Quickstart: Create a VM](#).
- **Required VM roles:**
  - Reader role on the virtual machine.
  - Reader role on the NIC with private IP of the virtual machine.
- **Required inbound ports:**
  - For Windows VMs - RDP (3389)
  - For Linux VMs - SSH (22)

### NOTE

The use of Azure Bastion with Azure Private DNS Zones is not supported at this time. Before you begin, please make sure that the virtual network where you plan to deploy your Bastion resource is not linked to a private DNS zone.

## Example values

You can use the following example values when creating this configuration, or you can substitute your own.

### Basic VNet and VM values:

NAME	VALUE
Virtual machine	TestVM
Resource group	TestRG1
Region	East US
Virtual network	VNet1
Address space	10.1.0.0/16
Subnets	FrontEnd: 10.1.0.0/24

#### Azure Bastion values:

NAME	VALUE
Name	VNet1-bastion
+ Subnet Name	AzureBastionSubnet
AzureBastionSubnet addresses	A subnet within your VNet address space with a subnet mask /26 or larger. For example, 10.1.1.0/26.
Tier/SKU	Standard
Instance count (host scaling)	3 or greater
Public IP address	Create new
Public IP address name	VNet1-ip
Public IP address SKU	Standard
Assignment	Static

#### IMPORTANT

For Azure Bastion resources deployed on or after November 2, 2021, the minimum AzureBastionSubnet size is /26 or larger (/25, /24, etc.). All Azure Bastion resources deployed in subnets of size /27 prior to this date are unaffected by this change and will continue to work, but we highly recommend increasing the size of any existing AzureBastionSubnet to /26 in case you choose to take advantage of [host scaling](#) in the future.

## Deploy Bastion

This section helps you deploy Bastion to your VNet. Once Bastion is deployed, you can connect securely to any VM in the VNet using its private IP address.

1. Sign in to the [Azure portal](#).
2. Go to your virtual network.

3. On the page for your virtual network, in the left pane, select **Bastion** to open the **Bastion** page.
4. On the Bastion page, select **Configure manually**. This lets you configure specific additional settings when deploying Bastion to your VNet.

Home > TestRG1 > VNet1

VNet1 | Bastion

Virtual network

Search (Ctrl+ /)

Diagnose and solve problems

**Settings**

- Address space
- Connected devices
- Subnets
- Bastion**
- DDoS protection
- Firewall
- Microsoft Defender for Cloud
- Network manager
- DNS servers
- Peerings

Azure Bastion protects your virtual machines by providing lightweight, browser-based connectivity without the need to expose them through public IP addresses. Deploying will automatically create a Bastion host on a subnet in your virtual network. [Learn more](#)

**Create Bastion**

Name ⓘ	VNet1-bastion
Resource group ⓘ	TestRG1
Virtual network ⓘ	VNet1
Public IP address ⓘ	VNet1-ip

Bastion pricing starts with an hourly base rate. [Learn more](#)

**Deploy Bastion**    **Configure manually**

5. On the **Create a Bastion** page, configure the settings for your bastion host. Project details are populated from your virtual network values. Configure the **Instance details** values.
  - **Name:** Type the name that you want to use for your bastion resource.
  - **Region:** The Azure public region in which the resource will be created. Choose the region in which your virtual network resides.
  - **Tier:** The tier is also known as the **SKU**. For this tutorial, select **Standard**. The Standard SKU lets you configure the instance count for host scaling and other features. For more information about features that require the Standard SKU, see [Configuration settings - SKU](#).
  - **Instance count:** This is the setting for **host scaling**. It's configured in scale unit increments. Use the slider or type a number to configure the instance count that you want. For this tutorial, you can select the instance count you'd prefer. For more information, see [Host scaling](#) and [Pricing](#).

**Instance details**

Name *	VNet1-bastion
Region *	East US
Tier *	Standard
Instance count *	3

6. Configure the **virtual networks** settings. Select your VNet from the dropdown. If you don't see your VNet in the dropdown list, make sure you selected the correct Region in the previous settings on this page.
7. To configure the AzureBastionSubnet, select **Manage subnet configuration**.

## Configure virtual networks

VNet1

Create new

To associate a virtual network with a Bastion, it must contain a subnet with name AzureBastionSubnet and a prefix of at least /26

Subnet \*

Manage subnet configuration

8. On the **Subnets** page, select **+ Subnet** to open the **Add subnet** page.
9. On the **Add subnet page**, create the 'AzureBastionSubnet' subnet using the following values. Leave the other values as default.
  - The subnet name must be **AzureBastionSubnet**.
  - The subnet must be at least **/26 or larger** (/26, /25, /24 etc.) to accommodate features available with the Standard SKU.Select **Save** at the bottom of the page to save your values.

10. At the top of the **Subnets** page, select **Create a Bastion** to return to the Bastion configuration page.

Home > TestRG1 > VNet1 | Bastion > **Create a Bastion** > VNet1

VNet1 | Subnets

Name	IPv4	IPv6	Available IPs	Delegated to
FrontEnd	10.1.0.0/24	-	249	-
AzureBastionSubnet	10.1.1.0/26	-	59	-

11. The **Public IP address** section is where you configure the public IP address of the Bastion host resource on which RDP/SSH will be accessed (over port 443). The public IP address must be in the same region as the Bastion resource you're creating. Create a new IP address. You can leave the default naming suggestion.
12. When you finish specifying the settings, select **Review + Create**. This validates the values.
13. Once validation passes, you can deploy Bastion. Select **Create**. You'll see a message letting you know that your deployment is in process. Status will display on this page as the resources are created. It takes about 10 minutes for the Bastion resource to be created and deployed.

## Connect to a VM

You can use any of the following detailed articles to connect to a VM. Some connection types require the Bastion **Standard SKU**.

- [Windows - RDP](#)
- [Windows - SSH](#)
- [Linux - SSH](#)
- [Linux - RDP](#)
- [Connect from a local computer using a native client](#)
- [Connect to a scale set](#)

You can also use the basic [Connection steps](#) in the section below to connect to your VM.

## Connection steps

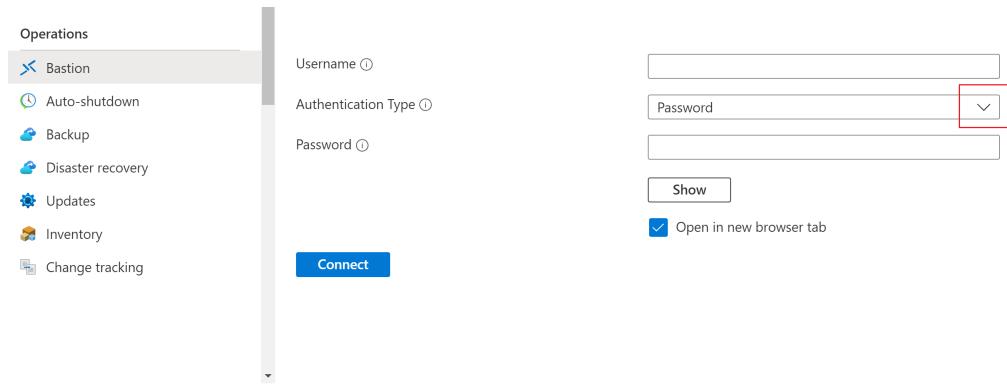
1. In the [Azure portal](#), go to the virtual machine to which you want to connect.
2. At the top of the page, select **Connect->Bastion** to go to the **Bastion** page. You can also go to the **Bastion** page using the left menu.
3. The options available on the **Bastion** page are dependant on the Bastion SKU tier. If you're using the **Basic SKU**, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22. You don't have options to change the port number or the protocol. However, you can change the keyboard language for RDP by expanding **Connection Settings**.

The screenshot shows the Azure portal interface for a 'TestVM' virtual machine. The left sidebar has 'Bastion' selected under 'Operations'. The main area displays the Bastion host information: 'Using Bastion: VNet1-bastion, Provisioning State: Succeeded'. It includes fields for 'Username', 'Authentication Type' (dropdown), 'Password', and a 'Show' button. A checked checkbox 'Open in new browser tab' is present. A large blue 'Connect' button is at the bottom. The 'Connection Settings' section is collapsed, indicated by a downward arrow icon.

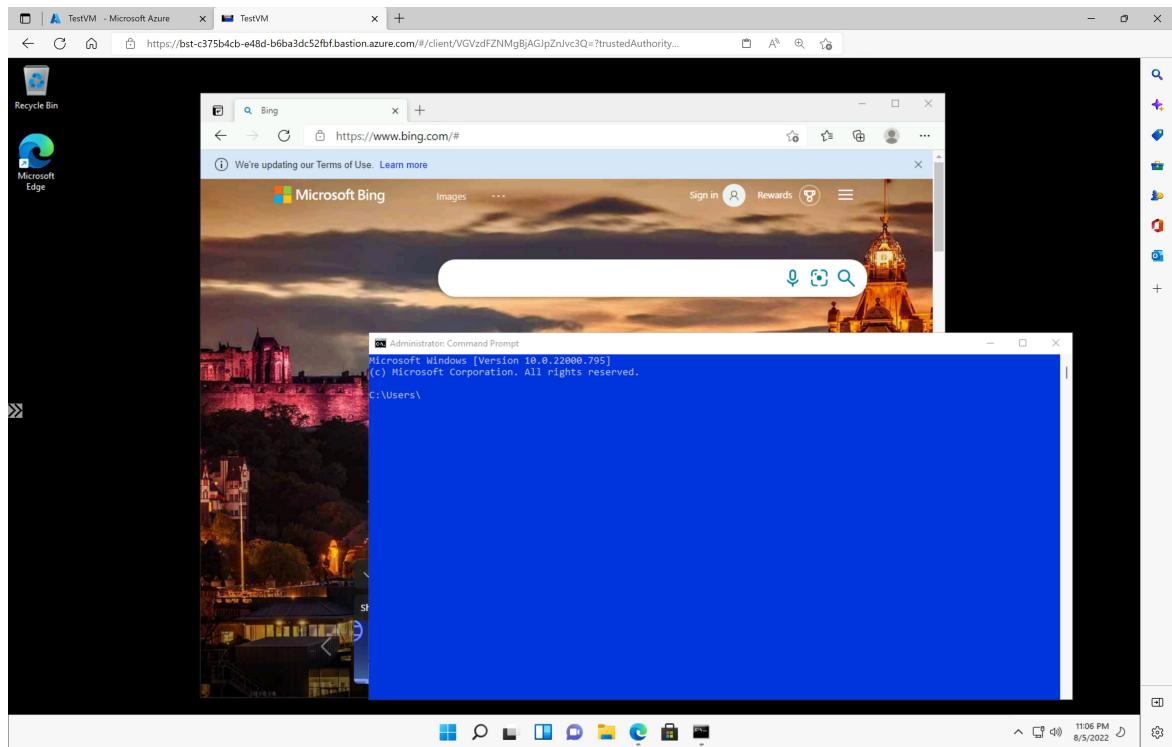
If you're using the **Standard SKU**, you have more connection protocol and port options available. Expand **Connection Settings** to see the options. Typically, unless you have configured different settings for your VM, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22.

The screenshot shows the same Azure portal interface for the 'TestVM' virtual machine, but with the 'Standard SKU' selected. The 'Connection Settings' section is now expanded, showing a radio button group for 'Protocol' (selected 'RDP') and 'SSH', a 'Port' field set to '3389' with a green checkmark, and a 'Keyboard Language' dropdown set to 'English (US)'.

4. Select the **Authentication Type** from the dropdown. The protocol determines the available authentication types. Complete the required authentication values.



5. To open the VM session in a new browser tab, leave **Open in a new browser tab** selected.
6. Click **Connect** to connect to the VM.
7. The connection to this virtual machine, via Bastion, will open directly in the Azure portal (over HTML5) using port 443 and the Bastion service.
  - When you connect, the desktop of the VM will look different than the example screenshot.
  - Using keyboard shortcut keys while connected to a VM may not result in the same behavior as shortcut keys on a local computer. For example, when connected to a Windows VM from a Windows client, CTRL+ALT+END is the keyboard shortcut for CTRL+ALT+Delete on a local computer. To do this from a Mac while connected to a Windows VM, the keyboard shortcut is Fn+CTRL+ALT+Backspace.



### To enable audio output

You can enable remote audio output for your VM. Some VMs automatically enable this setting, others require you to enable audio settings manually. The settings are changed on the VM itself. Your Bastion deployment doesn't need any special configuration settings to enable remote audio output.

#### NOTE

Audio output takes up bandwidth on your internet connection.

To enable remote audio output on a Windows VM:

1. After you're connected to the VM, on the right-hand bottom corner of the toolbar, you'll see an audio button.
2. Right-click the audio button and select "Sounds".
3. A pop-up appears asking if you would like to enable the Windows Audio Service. Select "Yes". You can configure more audio options in Sound preferences.
4. To verify sound output, hover your mouse over the audio button on the toolbar.

## Remove VM public IP address

When you connect to a VM using Azure Bastion, you don't need a public IP address for your VM. If you aren't using the public IP address for anything else, you can dissociate it from your VM. To dissociate a public IP address from your VM, use the following steps:

1. Go to your virtual machine and select **Networking**. Click the **NIC Public IP** to open the Public IP address page.

2. On the **Public IP address** page, you can see the VM network interface listed under **Associated to** on the lower right of the page. Click **Dissociate** at the top of the page.

3. Click **Yes** to dissociate the IP address from the network interface. Once the public IP address is dissociated from the VM network interface, you can see that it's no longer listed under **Associated to**.
4. After you dissociate the IP address, you can delete the public IP address resource. On the **Public IP address** page for the VM, select **Delete**.

The screenshot shows the Azure portal interface for managing a public IP address named 'TestVM-ip'. The left sidebar lists various navigation options like Overview, Activity log, Access control (IAM), Tags, Configuration, Properties, Locks, and Monitoring. The main content area is titled 'Essentials' and displays resource details: Resource group (TestRG1), Location (East US), Subscription (Content Development), Subscription ID, Tags, and Associated to. At the top right, there are buttons for Associate, Dissociate, Move, Delete (which is highlighted with a red box), Refresh, and a search bar.

5. Click **Yes** to delete the public IP address.

## Clean up resources

If you're not going to continue to use this application, delete your resources using the following steps:

1. Enter the name of your resource group in the **Search** box at the top of the portal. When you see your resource group in the search results, select it.
2. Select **Delete resource group**.
3. Enter the name of your resource group for **TYPE THE RESOURCE GROUP NAME:** and select **Delete**.

## Next steps

In this tutorial, you deployed Bastion to a virtual network and connected to a VM. You then removed the public IP address from the VM. Next, learn about and configure additional Bastion features.

[Bastion features and configuration settings](#)

[Bastion - VM connections and features](#)

# Deploy Bastion using Azure PowerShell

9/21/2022 • 8 minutes to read • [Edit Online](#)

This article shows you how to deploy Azure Bastion with the Standard SKU using PowerShell. Azure Bastion is a PaaS service that's maintained for you, not a bastion host that you install on your VM and maintain yourself. An Azure Bastion deployment is per virtual network, not per subscription/account or virtual machine. For more information about Azure Bastion, see [What is Azure Bastion?](#)

Once you deploy Bastion to your virtual network, you can connect to your VMs via private IP address. This seamless RDP/SSH experience is available to all the VMs in the same virtual network. If your VM has a public IP address that you don't need for anything else, you can remove it.

You can also deploy Bastion by using the following other methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Quickstart - deploy with default settings](#)

If you don't have an Azure subscription, create a [free account](#) before you begin.

## NOTE

The use of Azure Bastion with Azure Private DNS Zones is not supported at this time. Before you begin, please make sure that the virtual network where you plan to deploy your Bastion resource is not linked to a private DNS zone.

## Prerequisites

The following prerequisites are required.

### Azure PowerShell

This article uses PowerShell cmdlets. To run the cmdlets, you can use Azure Cloud Shell. Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open Cloud Shell, just select **Try it** from the upper-right corner of a code block. You can also open Cloud Shell on a separate browser tab by going to <https://shell.azure.com/powershell>. Select **Copy** to copy the blocks of code, paste them into Cloud Shell, and select the Enter key to run them.

You can also install and run the Azure PowerShell cmdlets locally on your computer. PowerShell cmdlets are updated frequently. If you have not installed the latest version, the values specified in the instructions may fail.

To find the versions of Azure PowerShell installed on your computer, use the `Get-Module -ListAvailable Az` cmdlet. To install or update, see [Install the Azure PowerShell module](#).

### Example values

You can use the following example values when creating this configuration, or you can substitute your own.

#### Basic VNet and VM values:

NAME	VALUE
Virtual machine	TestVM

NAME	VALUE
Resource group	TestRG1
Region	East US
Virtual network	VNet1
Address space	10.1.0.0/16
Subnets	FrontEnd: 10.1.0.0/24

#### Azure Bastion values:

NAME	VALUE
Name	VNet1-bastion
Subnet Name	FrontEnd
Subnet Name	AzureBastionSubnet
AzureBastionSubnet addresses	A subnet within your VNet address space with a subnet mask /26 or larger. For example, 10.1.1.0/26.
Tier/SKU	Standard
Public IP address	Create new
Public IP address name	VNet1-ip
Public IP address SKU	Standard
Assignment	Static

## Deploy Bastion

This section helps you create a virtual network, subnets, and deploy Azure Bastion using Azure PowerShell.

1. Create an Azure resource group with [New-AzResourceGroup](#). A resource group is a logical container into which Azure resources are deployed and managed. If you're running PowerShell locally, open your PowerShell console with elevated privileges and connect to Azure using the [Connect-AzAccount](#) command.

```
New-AzResourceGroup -Name TestRG1 -Location EastUS
```

2. Create a virtual network.

```
$virtualNetwork = New-AzVirtualNetwork `  
-ResourceGroupName TestRG1 `  
-Location EastUS `  
-Name VNet1 `  
-AddressPrefix 10.1.0.0/16
```

3. Set the configuration for the virtual network.

```
$virtualNetwork | Set-AzVirtualNetwork
```

4. Configure and set a subnet for your virtual network. This will be the subnet to which you'll deploy a VM. The variable used for `-VirtualNetwork` was set in the previous steps.

```
$subnetConfig = Add-AzVirtualNetworkSubnetConfig `  
-Name 'FrontEnd' `  
-AddressPrefix 10.1.0.0/24 `  
-VirtualNetwork $virtualNetwork
```

```
$virtualNetwork | Set-AzVirtualNetwork
```

5. Configure and set the Azure Bastion subnet for your virtual network. This subnet is reserved exclusively for Azure Bastion resources. You must create the Azure Bastion subnet using the name value `AzureBastionSubnet`. This value lets Azure know which subnet to deploy the Bastion resources to. The example below also helps you add an Azure Bastion subnet to an existing VNet.

**IMPORTANT**

For Azure Bastion resources deployed on or after November 2, 2021, the minimum AzureBastionSubnet size is /26 or larger (/25, /24, etc.). All Azure Bastion resources deployed in subnets of size /27 prior to this date are unaffected by this change and will continue to work, but we highly recommend increasing the size of any existing AzureBastionSubnet to /26 in case you choose to take advantage of [host scaling](#) in the future.

- The smallest subnet AzureBastionSubnet size you can create is /26. We recommend that you create a /26 or larger size to accommodate host scaling.
  - For more information about scaling, see [Configuration settings - Host scaling](#).
  - For more information about settings, see [Configuration settings - AzureBastionSubnet](#).
- Create the `AzureBastionSubnet` without any route tables or delegations.
- If you use Network Security Groups on the `AzureBastionSubnet`, refer to the [Work with NSGs](#) article.

Declare the variable.

```
$virtualNetwork = Get-AzVirtualNetwork -Name "VNet1" `  
-ResourceGroupName "TestRG1"
```

Add the configuration.

```
Add-AzVirtualNetworkSubnetConfig -Name "AzureBastionSubnet" `  
-VirtualNetwork $virtualNetwork -AddressPrefix "10.1.1.0/26" `
```

Set the configuration.

```
$virtualNetwork | Set-AzVirtualNetwork
```

6. Create a public IP address for Azure Bastion. The public IP is the public IP address the Bastion resource on which RDP/SSH will be accessed (over port 443). The public IP address must be in the same region as the Bastion resource you're creating.

```
$publicip = New-AzPublicIpAddress -ResourceGroupName "TestRG1" -name "VNet1-ip" -location "EastUS"  
-AllocationMethod Static -Sku Standard
```

7. Create a new Azure Bastion resource in the AzureBastionSubnet using the [New-AzBastion](#) command. The following example uses the **Standard SKU**. The Standard SKU lets you configure more Bastion features and connect to VMs using more connection types. For more information, see [Bastion SKUs](#). If you want to deploy using the Basic SKU, change the -Sku value to "Basic".

```
New-AzBastion -ResourceGroupName "TestRG1" -Name "VNet1-bastion" `  
-PublicIpAddressRgName "TestRG1" -PublicIpAddressName "VNet1-ip" `  
-VirtualNetworkRgName "TestRG1" -VirtualNetworkName "VNet1" `  
-Sku "Standard"
```

8. It takes about 10 minutes for the Bastion resources to deploy. You can create a VM in the next section while Bastion deploys to your virtual network.

## Create a VM

You can create a VM using the [Quickstart: Create a VM using PowerShell](#) or [Quickstart: Create a VM using the portal](#) articles. Be sure you deploy the VM to the virtual network to which you deployed Bastion. The VM you create in this section isn't a part of the Bastion configuration and doesn't become a bastion host. You connect to this VM later in this tutorial via Bastion.

The following required roles for your resources.

- Required VM roles:
  - Reader role on the virtual machine.
  - Reader role on the NIC with private IP of the virtual machine.
- Required inbound ports:
  - For Windows VMS - RDP (3389)
  - For Linux VMs - SSH (22)

## Connect to a VM

You can use the [Connection steps](#) in the section below to connect to your VM. You can also use any of the following articles to connect to a VM. Some connection types require the Bastion [Standard SKU](#).

- [Windows - RDP](#)
- [Windows - SSH](#)
- [Linux - SSH](#)
- [Linux - RDP](#)
- [Connect from a local computer using a native client](#)
- [Connect to a scale set](#)

### Connection steps

1. In the [Azure portal](#), go to the virtual machine to which you want to connect.
2. At the top of the page, select **Connect->Bastion** to go to the **Bastion** page. You can also go to the **Bastion** page using the left menu.
3. The options available on the **Bastion** page are dependant on the Bastion SKU tier. If you're using the **Basic SKU**, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22. You don't have options to change the port number or the protocol. However, you can change the keyboard language for RDP by expanding **Connection Settings**.

Home > TestVM

## TestVM | Bastion

Virtual machine

Search (Ctrl+ /)

Availability + scaling

Configuration

Identity

Properties

Locks

**Operations**

- Bastion** (selected)
- Auto-shutdown
- Backup
- Disaster recovery
- Updates
- Inventory
- Change tracking

Azure Bastion protects your virtual machines by providing lightweight, browser-based connectivity without the need to expose them through public IP addresses. Deploying will automatically create a Bastion host on a subnet in your virtual network. [Learn more](#)

Using Bastion: **VNet1-bastion**, Provisioning State: **Succeeded**

Please enter username and password to your virtual machine to connect using Bastion.

**Connection Settings**

Username

Authentication Type  RDP  SSH

Password

Show

Open in new browser tab

**Connect**

If you're using the **Standard SKU**, you have more connection protocol and port options available. Expand **Connection Settings** to see the options. Typically, unless you have configured different settings for your VM, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22.

Home > TestVM

## TestVM | Bastion

Virtual machine

Search (Ctrl+ /)

Availability + scaling

Configuration

Identity

Properties

Locks

**Operations**

- Bastion** (selected)
- Auto-shutdown
- Backup
- Disaster recovery
- Updates
- Inventory
- Change tracking

Azure Bastion protects your virtual machines by providing lightweight, browser-based connectivity without the need to expose them through public IP addresses. Deploying will automatically create a Bastion host on a subnet in your virtual network. [Learn more](#)

Using Bastion: **VNet1-bastion**, Provisioning State: **Succeeded**

Please enter username and password to your virtual machine to connect using Bastion.

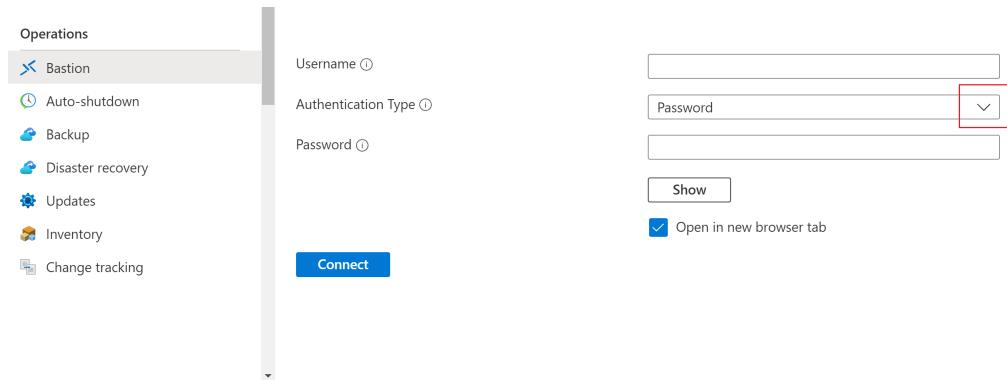
**Connection Settings**

Protocol \*  RDP  SSH

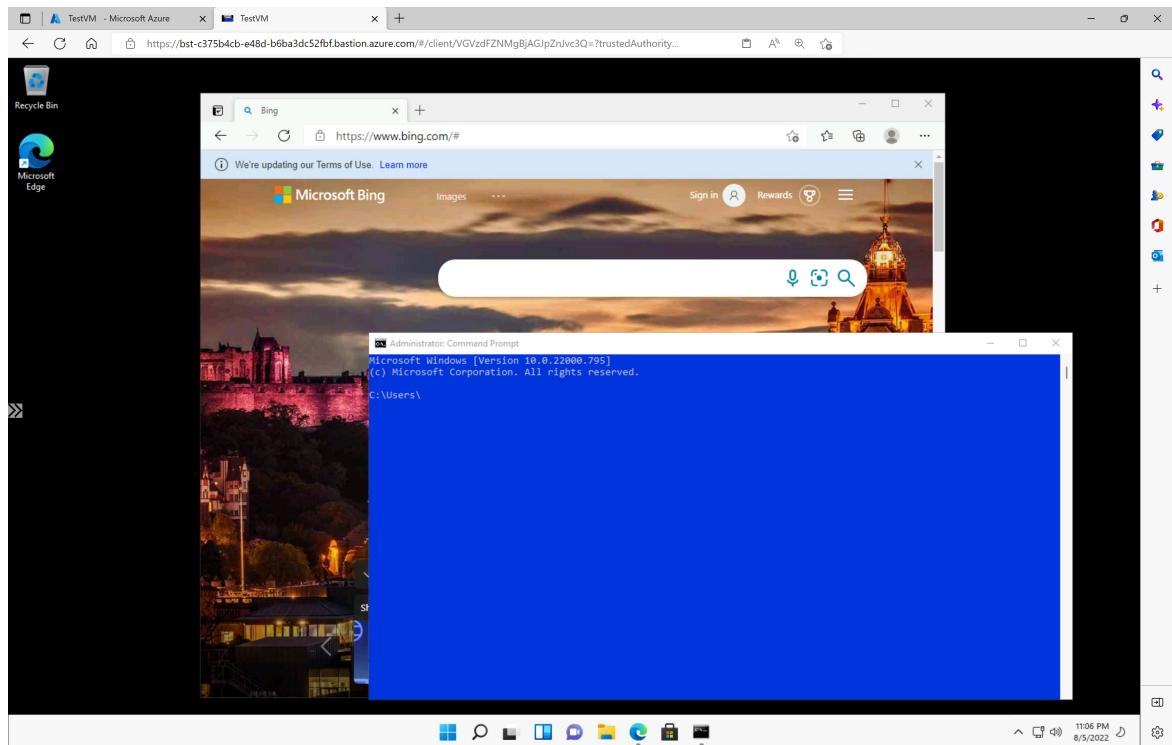
Port

Keyboard Language

4. Select the **Authentication Type** from the dropdown. The protocol determines the available authentication types. Complete the required authentication values.



5. To open the VM session in a new browser tab, leave **Open in a new browser tab** selected.
6. Click **Connect** to connect to the VM.
7. The connection to this virtual machine, via Bastion, will open directly in the Azure portal (over HTML5) using port 443 and the Bastion service.
  - When you connect, the desktop of the VM will look different than the example screenshot.
  - Using keyboard shortcut keys while connected to a VM may not result in the same behavior as shortcut keys on a local computer. For example, when connected to a Windows VM from a Windows client, CTRL+ALT+END is the keyboard shortcut for CTRL+ALT+Delete on a local computer. To do this from a Mac while connected to a Windows VM, the keyboard shortcut is Fn+CTRL+ALT+Backspace.



### To enable audio output

You can enable remote audio output for your VM. Some VMs automatically enable this setting, others require you to enable audio settings manually. The settings are changed on the VM itself. Your Bastion deployment doesn't need any special configuration settings to enable remote audio output.

#### NOTE

Audio output takes up bandwidth on your internet connection.

To enable remote audio output on a Windows VM:

1. After you're connected to the VM, on the right-hand bottom corner of the toolbar, you'll see an audio button.
2. Right-click the audio button and select "Sounds".
3. A pop-up appears asking if you would like to enable the Windows Audio Service. Select "Yes". You can configure more audio options in Sound preferences.
4. To verify sound output, hover your mouse over the audio button on the toolbar.

## Remove VM public IP address

Azure Bastion doesn't use the public IP address to connect to the client VM. If you don't need the public IP address for your VM, you can disassociate the public IP address. See [Dissociate a public IP address from an Azure VM](#).

## Next steps

- To use Network Security Groups with the Azure Bastion subnet, see [Work with NSGs](#).
- To understand VNet peering, see [VNet peering and Azure Bastion](#).

# Deploy Bastion using Azure CLI

9/21/2022 • 6 minutes to read • [Edit Online](#)

This article shows you how to deploy Azure Bastion using CLI. Azure Bastion is a PaaS service that's maintained for you, not a bastion host that you install on your VM and maintain yourself. An Azure Bastion deployment is per virtual network, not per subscription/account or virtual machine. For more information about Azure Bastion, see [What is Azure Bastion?](#)

Once you deploy Bastion to your virtual network, you can connect to your VMs via private IP address. This seamless RDP/SSH experience is available to all the VMs in the same virtual network. If your VM has a public IP address that you don't need for anything else, you can remove it.

You can also deploy Bastion by using the following other methods:

- [Azure portal](#)
- [Azure PowerShell](#)
- [Quickstart - deploy with default settings](#)

## Prerequisites

### Azure subscription

Verify that you have an Azure subscription. If you don't already have an Azure subscription, you can activate your [MSDN subscriber benefits](#) or sign up for a [free account](#).

### Azure CLI

This article uses the Azure CLI. To run commands, you can use Azure Cloud Shell. The Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your account.

To open the Cloud Shell, just select **Try it** from the upper right corner of a code block. You can also launch Cloud Shell in a separate browser tab by going to <https://shell.azure.com> and toggle the dropdown in the left corner to reflect Bash or PowerShell. Select **Copy** to copy the blocks of code, paste it into the Cloud Shell, and press enter to run it.

#### NOTE

The use of Azure Bastion with Azure Private DNS Zones is not supported at this time. Before you begin, please make sure that the virtual network where you plan to deploy your Bastion resource is not linked to a private DNS zone.

## Deploy Bastion

This section helps you deploy Azure Bastion using Azure CLI.

#### NOTE

As shown in the examples, use the `--location` parameter with `--resource-group` for every command to ensure that the resources are deployed together.

1. Create a virtual network and an Azure Bastion subnet. You must create the Azure Bastion subnet using the name value **AzureBastionSubnet**. This value lets Azure know which subnet to deploy the Bastion

resources to. This is different than a VPN gateway subnet.

#### IMPORTANT

For Azure Bastion resources deployed on or after November 2, 2021, the minimum AzureBastionSubnet size is /26 or larger (/25, /24, etc.). All Azure Bastion resources deployed in subnets of size /27 prior to this date are unaffected by this change and will continue to work, but we highly recommend increasing the size of any existing AzureBastionSubnet to /26 in case you choose to take advantage of [host scaling](#) in the future.

- The smallest subnet AzureBastionSubnet size you can create is /26. We recommend that you create a /26 or larger size to accommodate host scaling.
  - For more information about scaling, see [Configuration settings - Host scaling](#).
  - For more information about settings, see [Configuration settings - AzureBastionSubnet](#).
- Create the **AzureBastionSubnet** without any route tables or delegations.
- If you use Network Security Groups on the **AzureBastionSubnet**, refer to the [Work with NSGs](#) article.

```
az network vnet create --resource-group MyResourceGroup --name MyVnet --address-prefix 10.0.0.0/16 --subnet-name AzureBastionSubnet --subnet-prefix 10.0.0.0/24 --location northeurope
```

2. Create a public IP address for Azure Bastion. The public IP is the public IP address the Bastion resource on which RDP/SSH will be accessed (over port 443). The public IP address must be in the same region as the Bastion resource you're creating.

The following example uses the **Standard SKU**. The Standard SKU lets you configure more Bastion features and connect to VMs using more connection types. For more information, see [Bastion SKUs](#).

```
az network public-ip create --resource-group MyResourceGroup --name MyIp --sku Standard --location northeurope
```

3. Create a new Azure Bastion resource in the AzureBastionSubnet of your virtual network. It takes about 10 minutes for the Bastion resource to create and deploy.

```
az network bastion create --name MyBastion --public-ip-address MyIp --resource-group MyResourceGroup --vnet-name MyVnet --location northeurope
```

## Connect to a VM

You can use the [Connection steps](#) in the section below to connect to your VM. You can also use any of the following articles to connect to a VM. Some connection types require the Bastion [Standard SKU](#).

- [Windows - RDP](#)
- [Windows - SSH](#)
- [Linux - SSH](#)
- [Linux - RDP](#)
- [Connect from a local computer using a native client](#)
- [Connect to a scale set](#)

### Connection steps

1. In the [Azure portal](#), go to the virtual machine to which you want to connect.
2. At the top of the page, select **Connect->Bastion** to go to the **Bastion** page. You can also go to the

Bastion page using the left menu.

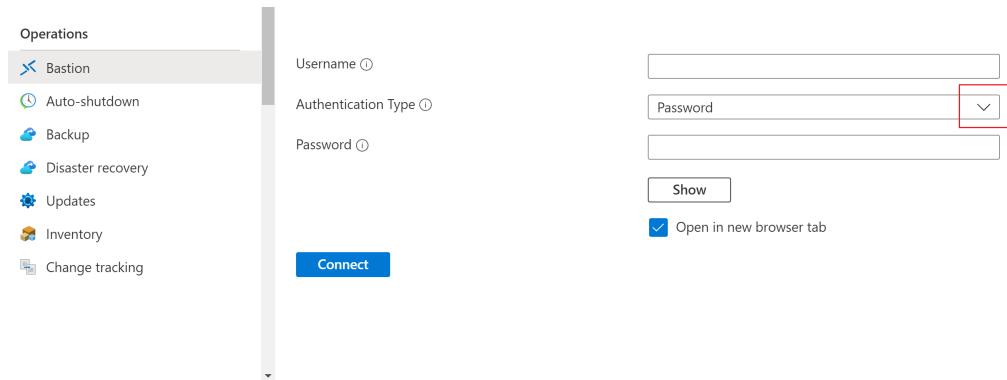
3. The options available on the **Bastion** page are dependant on the Bastion SKU tier. If you're using the **Basic SKU**, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22. You don't have options to change the port number or the protocol. However, you can change the keyboard language for RDP by expanding **Connection Settings**.

The screenshot shows the 'TestVM | Bastion' page. The left sidebar has a 'Bastion' section selected. The main area displays a message about Azure Bastion protecting virtual machines. Below it, it says 'Using Bastion: VNet1-bastion, Provisioning State: Succeeded'. A note asks to enter a username and password. Under 'Connection Settings', there are fields for 'Username', 'Authentication Type' (dropdown), 'Password', and a 'Show' button. A checked checkbox says 'Open in new browser tab'. At the bottom is a 'Connect' button.

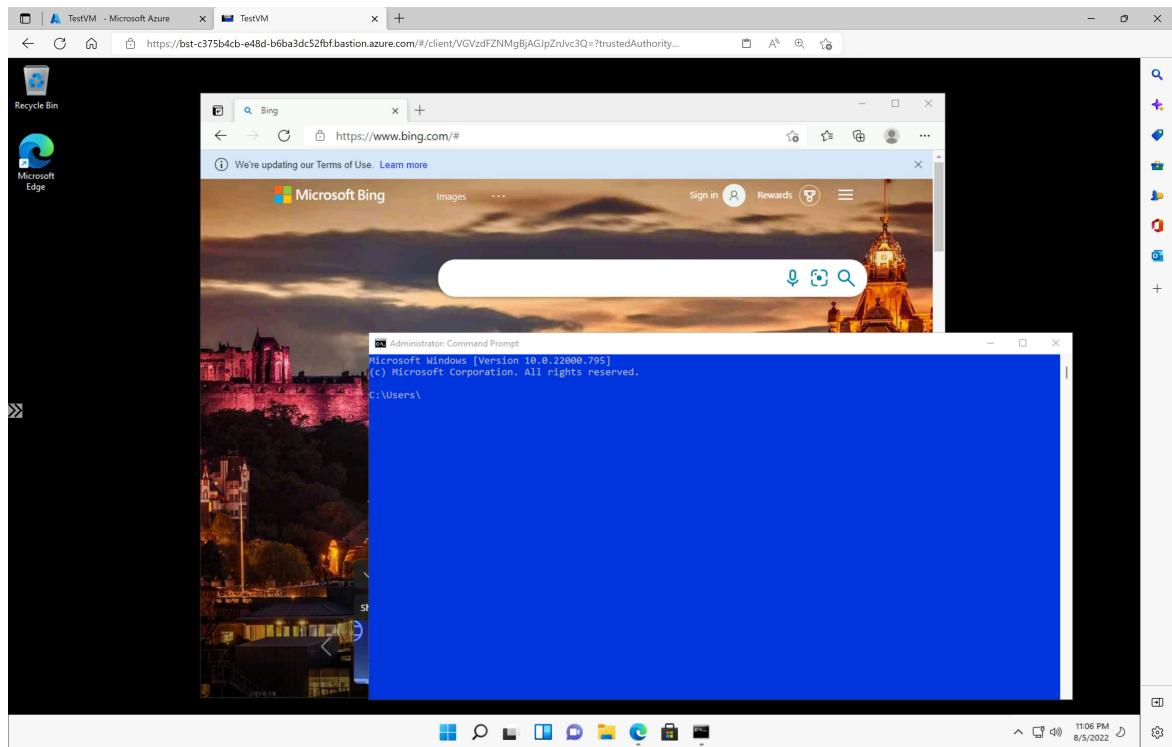
If you're using the **Standard SKU**, you have more connection protocol and port options available. Expand **Connection Settings** to see the options. Typically, unless you have configured different settings for your VM, you connect to a Windows computer using RDP and port 3389, and to a Linux computer using SSH and port 22.

The screenshot shows the same 'TestVM | Bastion' page, but the 'Bastion' section in the sidebar is not selected. The 'Connection Settings' section is expanded, showing 'Protocol' (radio buttons for 'RDP' and 'SSH', with 'RDP' selected), 'Port' (text input '3389'), and 'Keyboard Language' (dropdown 'English (US)').

4. Select the **Authentication Type** from the dropdown. The protocol determines the available authentication types. Complete the required authentication values.



5. To open the VM session in a new browser tab, leave **Open in a new browser tab** selected.
6. Click **Connect** to connect to the VM.
7. The connection to this virtual machine, via Bastion, will open directly in the Azure portal (over HTML5) using port 443 and the Bastion service.
  - When you connect, the desktop of the VM will look different than the example screenshot.
  - Using keyboard shortcut keys while connected to a VM may not result in the same behavior as shortcut keys on a local computer. For example, when connected to a Windows VM from a Windows client, CTRL+ALT+END is the keyboard shortcut for CTRL+ALT+Delete on a local computer. To do this from a Mac while connected to a Windows VM, the keyboard shortcut is Fn+CTRL+ALT+Backspace.



### To enable audio output

You can enable remote audio output for your VM. Some VMs automatically enable this setting, others require you to enable audio settings manually. The settings are changed on the VM itself. Your Bastion deployment doesn't need any special configuration settings to enable remote audio output.

#### NOTE

Audio output takes up bandwidth on your internet connection.

To enable remote audio output on a Windows VM:

1. After you're connected to the VM, on the right-hand bottom corner of the toolbar, you'll see an audio button.
2. Right-click the audio button and select "Sounds".
3. A pop-up appears asking if you would like to enable the Windows Audio Service. Select "Yes". You can configure more audio options in Sound preferences.
4. To verify sound output, hover your mouse over the audio button on the toolbar.

## Remove VM public IP address

Azure Bastion doesn't use the public IP address to connect to the client VM. If you don't need the public IP address for your VM, you can disassociate the public IP address. See [Dissociate a public IP address from an Azure VM](#).

## Next steps

- To use Network Security Groups with the Azure Bastion subnet, see [Work with NSGs](#).
- To understand VNet peering, see [VNet peering and Azure Bastion](#).

# Create an SSH connection to a Linux VM using Azure Bastion

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article shows you how to securely and seamlessly create an SSH connection to your Linux VMs located in an Azure virtual network directly through the Azure portal. When you use Azure Bastion, your VMs don't require a client, agent, or additional software. You can also connect to a Linux VM using RDP. For information, see [Create an RDP connection to a Linux VM](#).

Azure Bastion provides secure connectivity to all of the VMs in the virtual network in which it's provisioned. Using Azure Bastion protects your virtual machines from exposing RDP/SSH ports to the outside world, while still providing secure access using RDP/SSH. For more information, see the [What is Azure Bastion?](#) overview article.

When connecting to a Linux virtual machine using SSH, you can use both username/password and SSH keys for authentication.

The SSH private key must be in a format that begins with `"-----BEGIN RSA PRIVATE KEY-----"` and ends with `"-----END RSA PRIVATE KEY-----"`.

## Prerequisites

Make sure that you have set up an Azure Bastion host for the virtual network in which the VM resides. For more information, see [Create an Azure Bastion host](#). Once the Bastion service is provisioned and deployed in your virtual network, you can use it to connect to any VM in this virtual network.

### Required roles

In order to make a connection, the following roles are required:

- Reader role on the virtual machine
- Reader role on the NIC with private IP of the virtual machine
- Reader role on the Azure Bastion resource

### Ports

In order to connect to the Linux VM via SSH, you must have the following ports open on your VM:

- Inbound port: SSH (22) *or*
- Inbound port: Custom value (you'll then need to specify this custom port when you connect to the VM via Azure Bastion)

#### NOTE

If you want to specify a custom port value, Azure Bastion must be configured using the Standard SKU. The Basic SKU does not allow you to specify custom ports.

## Bastion connection page

1. In the [Azure portal](#), go to the virtual machine that you want to connect to. On the [Overview](#) page, select **Connect**, then select **Bastion** from the dropdown to open the Bastion connection page. You can also

select **Bastion** from the left pane.

The screenshot shows the Azure portal interface for a virtual machine named "TestVM". The left sidebar has a "Bastion" section selected. The main content area shows basic VM details: Status (Running), Location (East US), Subscription (Content Development), and Tags (Click here to add tags). At the top, there's a "Connect" dropdown menu with options: SSH, RDP, and Bastion (which is highlighted with a red box).

2. On the **Bastion** connection page, click the **Connection Settings** arrow to expand all the available settings. If you are using a Bastion **Standard** SKU, you have more available settings than a Basic SKU.

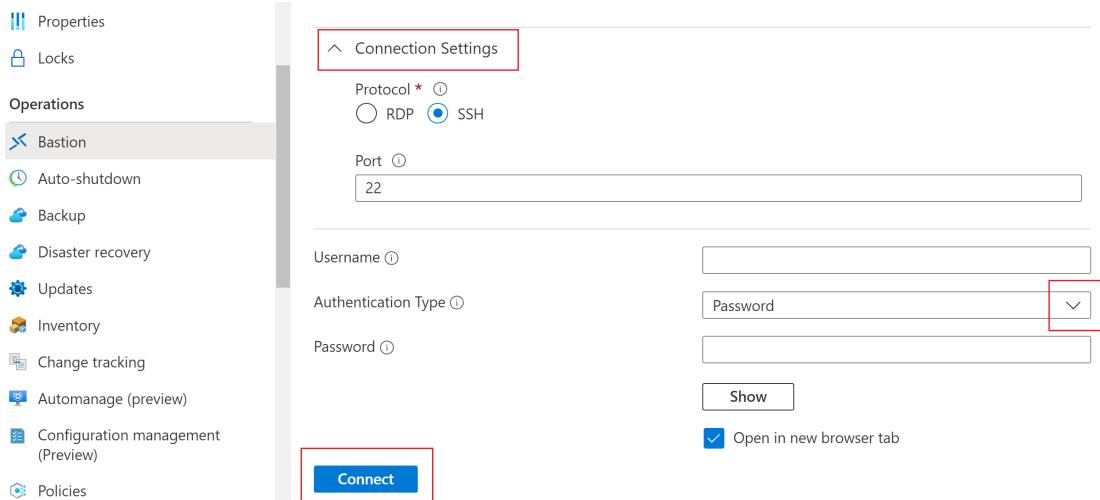
The screenshot shows the "TestVM | Bastion" page. The "Bastion" section in the left sidebar is selected. In the main content area, under "Connection Settings", the "Protocol" dropdown is set to "SSH" (radio button selected) and the "Port" input field contains "22". A note above says: "Please enter username and password to your virtual machine to connect using Bastion."

3. Authenticate and connect using one of the methods in the following sections.

- [Username and password](#)
- [Private key from local file](#)
- [Password - Azure Key Vault](#)
- [Private key - Azure Key Vault](#)

## Username and password

Use the following steps to authenticate using username and password.



1. To authenticate using a username and password, configure the following settings:

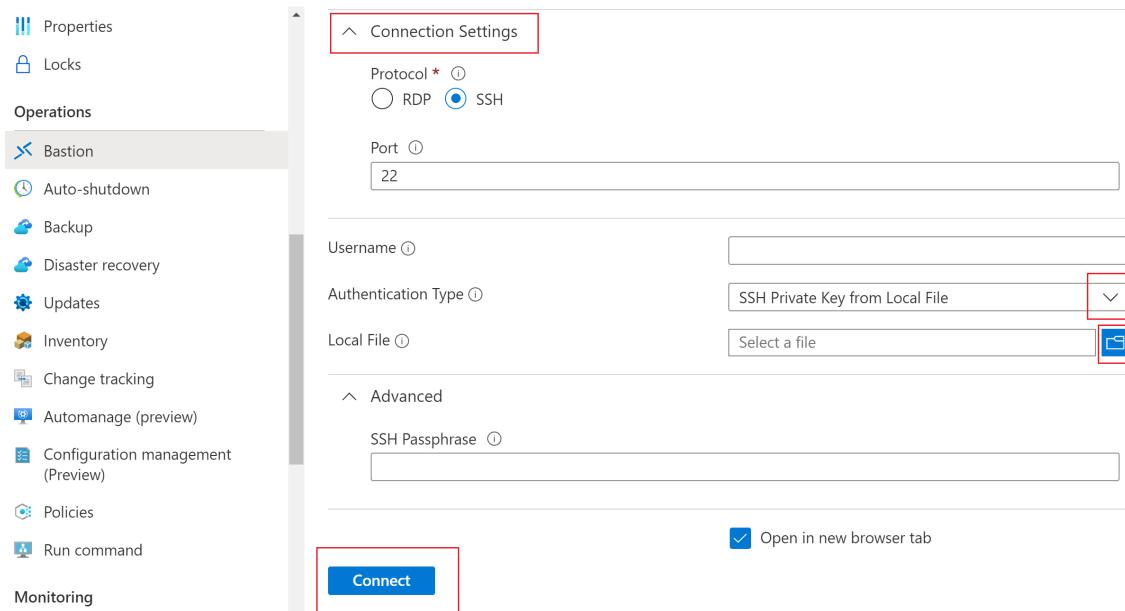
- **Protocol:** Select SSH.
- **Port:** Input the port number. Custom port connections are available for the Standard SKU only.
- **Authentication type:** Select **Password** from the dropdown.
- **Username:** Enter the username.
- **Password:** Enter the **Password**.

2. To work with the VM in a new browser tab, select **Open in new browser tab**.

3. Click **Connect** to connect to the VM.

## Private key from local file

Use the following steps to authenticate using an SSH private key from a local file.



1. To authenticate using a private key from a local file, configure the following settings:

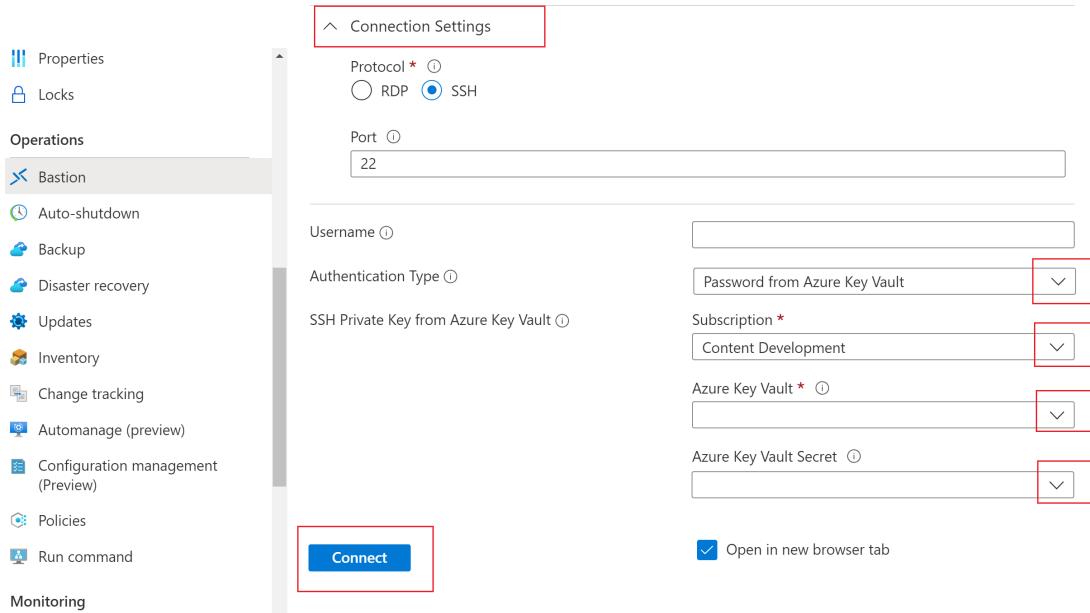
- **Protocol:** Select SSH.
- **Port:** Input the port number. Custom port connections are available for the Standard SKU only.
- **Authentication type:** Select **SSH Private Key from Local File** from the dropdown.
- **Local File:** Select the local file.
- **SSH Passphrase:** Enter the SSH passphrase if necessary.

2. To work with the VM in a new browser tab, select **Open in new browser tab**.

3. Click **Connect** to connect to the VM.

## Password - Azure Key Vault

Use the following steps to authenticate using a password from Azure Key Vault.



1. To authenticate using a password from Azure Key Vault, configure the following settings:

- **Protocol:** Select SSH.
- **Port:** Input the port number. Custom port connections are available for the Standard SKU only.
- **Authentication type:** Select **Password from Azure Key Vault** from the dropdown.
- **Username:** Enter the username.
- **Subscription:** Select the subscription.
- **Azure Key Vault:** Select the Key Vault.
- **Azure Key Vault Secret:** Select the Key Vault secret containing the value of your SSH private key.
  - If you didn't set up an Azure Key Vault resource, see [Create a key vault](#) and store your SSH private key as the value of a new Key Vault secret.
  - Make sure you have **List** and **Get** access to the secrets stored in the Key Vault resource. To assign and modify access policies for your Key Vault resource, see [Assign a Key Vault access policy](#).

### NOTE

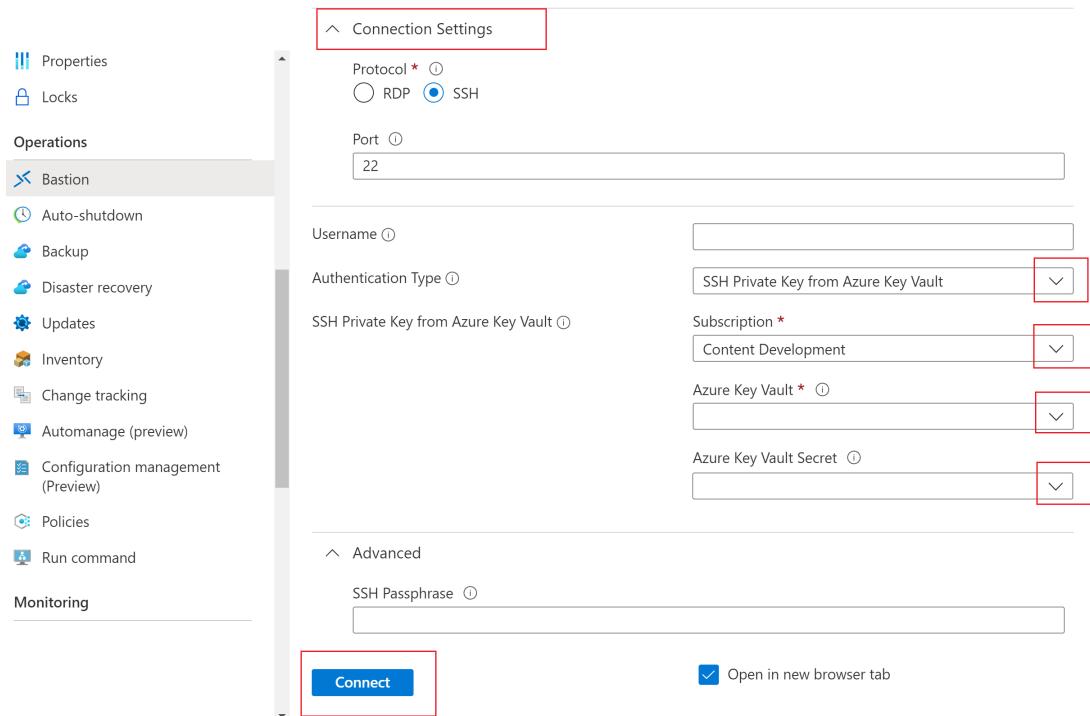
Please store your SSH private key as a secret in Azure Key Vault using the **PowerShell** or **Azure CLI** experience. Storing your private key via the Azure Key Vault portal experience will interfere with the formatting and result in unsuccessful login. If you did store your private key as a secret using the portal experience and no longer have access to the original private key file, see [Update SSH key](#) to update access to your target VM with a new SSH key pair.

2. To work with the VM in a new browser tab, select **Open in new browser tab**.

3. Click **Connect** to connect to the VM.

# Private key - Azure Key Vault

Use the following steps to authenticate using a private key stored in Azure Key Vault.



1. To authenticate using a private key stored in Azure Key Vault, configure the following settings:

- **Protocol:** Select SSH.
- **Port:** Input the port number. Custom port connections are available for the Standard SKU only.
- **Authentication type:** Select **SSH Private Key from Azure Key Vault** from the dropdown.
- **Username:** Enter the username.
- **Subscription:** Select the subscription.
- **Azure Key Vault:** Select the Key Vault.
  - If you didn't set up an Azure Key Vault resource, see [Create a key vault](#) and store your SSH private key as the value of a new Key Vault secret.
  - Make sure you have **List** and **Get** access to the secrets stored in the Key Vault resource. To assign and modify access policies for your Key Vault resource, see [Assign a Key Vault access policy](#).

#### NOTE

Please store your SSH private key as a secret in Azure Key Vault using the **PowerShell** or **Azure CLI** experience. Storing your private key via the Azure Key Vault portal experience will interfere with the formatting and result in unsuccessful login. If you did store your private key as a secret using the portal experience and no longer have access to the original private key file, see [Update SSH key](#) to update access to your target VM with a new SSH key pair.

- **Azure Key Vault Secret:** Select the Key Vault secret containing the value of your SSH private key.
2. To work with the VM in a new browser tab, select **Open in new browser tab**.
3. Click **Connect** to connect to the VM.

## Next steps

For more information about Azure Bastion, see the [Bastion FAQ](#).

# Create an RDP connection to a Windows VM using Azure Bastion

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article shows you how to securely and seamlessly create an RDP connection to your Windows VMs located in an Azure virtual network directly through the Azure portal. When you use Azure Bastion, your VMs don't require a client, agent, or additional software. You can also connect to a Windows VM using SSH. For information, see [Create an SSH connection to a Windows VM](#).

Azure Bastion provides secure connectivity to all of the VMs in the virtual network in which it's provisioned. Using Azure Bastion protects your virtual machines from exposing RDP/SSH ports to the outside world, while still providing secure access using RDP/SSH. For more information, see [What is Azure Bastion?](#)

## Prerequisites

Before you begin, verify that you've met the following criteria:

- A VNet with the Bastion host already installed.
  - Make sure that you have set up an Azure Bastion host for the virtual network in which the VM is located. Once the Bastion service is provisioned and deployed in your virtual network, you can use it to connect to any VM in the virtual network.
  - To set up an Azure Bastion host, see [Create a bastion host](#). If you plan to configure custom port values, be sure to select the Standard SKU when configuring Bastion.
- A Windows virtual machine in the virtual network.

## Required roles

- Reader role on the virtual machine.
- Reader role on the NIC with private IP of the virtual machine.
- Reader role on the Azure Bastion resource.

## Ports

To connect to the Windows VM, you must have the following ports open on your Windows VM:

- Inbound port: RDP (3389) *or*
- Inbound port: Custom value (you'll then need to specify this custom port when you connect to the VM via Azure Bastion)

### NOTE

If you want to specify a custom port value, Azure Bastion must be configured using the Standard SKU. The Basic SKU does not allow you to specify custom ports.

## Connect

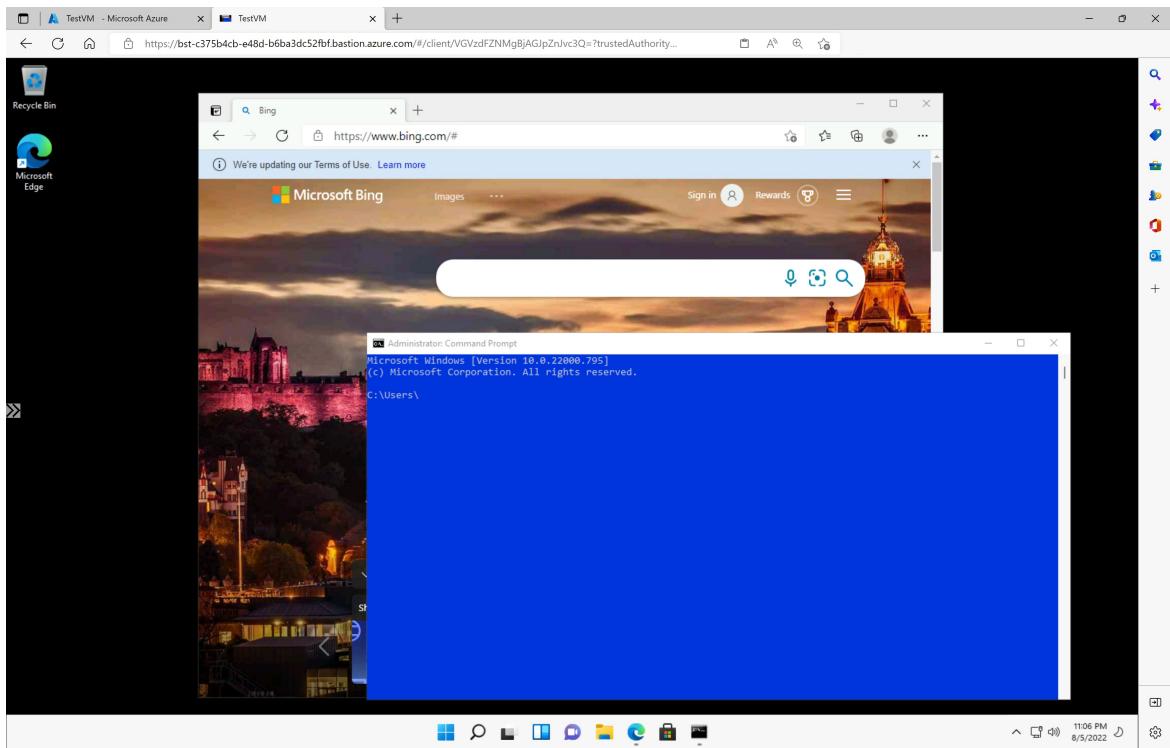
1. In the [Azure portal](#), go to the virtual machine that you want to connect to. On the **Overview** page, select **Connect**, then select **Bastion** from the dropdown to open the Bastion page. You can also select **Bastion** from the left pane.

The screenshot shows the Azure portal interface for a virtual machine named "TestVM". The left sidebar lists various operations like Availability + scaling, Configuration, Identity, Properties, Locks, and Bastion. The "Bastion" option is highlighted with a red box. The main content area displays the VM's properties: Resource group (TestRG1), Status (Running), Location (East US), Subscription (move), Operating system (Windows (Windows 11 Pro)), Size (Standard D2s v3 (2 vcpus, 8 GiB memory)), Public IP address, and Virtual network/subnet (VNet1/FrontEnd). A callout box points to the "Bastion" section in the top navigation bar.

2. On the **Bastion** page, enter the required authentication credentials, then click **Connect**. If you configured your bastion host using the Standard SKU, you'll see additional credential options on this page. If your VM is domain-joined, you must use the following format: **username@domain.com**.

The screenshot shows the "Bastion" page for "TestVM". The left sidebar includes options like Availability + scaling, Configuration, Identity, Properties, Locks, and Bastion (which is selected and highlighted with a red box). The main content area contains a brief description of Azure Bastion Service, provisioning state information (Using Bastion: VNet1-bastion, Provisioning State: Succeeded), and a form for entering connection details. The form includes fields for Username, Authentication Type (set to Password), Password, and a "Show" password toggle. A checkbox for "Open in new browser tab" is checked. A "Connect" button is at the bottom of the form.

3. When you click **Connect**, the RDP connection to this virtual machine via Bastion will open in your browser (over HTML5) using port 443 and the Bastion service. The following example shows a connection to a Windows 11 virtual machine in a new browser tab. The page you see depends on the VM you're connecting to.



When working with the VM, using keyboard shortcut keys may not result in the same behavior as shortcut keys on a local computer. For example, when connected to a Windows VM from a Windows client, CTRL+ALT+END is the keyboard shortcut for CTRL+ALT+Delete on a local computer. To do this from a Mac while connected to a Windows VM, the keyboard shortcut is Fn+CTRL+ALT+Backspace.

## Next steps

Read the [Bastion FAQ](#).

# What is Azure role-based access control (Azure RBAC)?

9/21/2022 • 6 minutes to read • [Edit Online](#)

Access management for cloud resources is a critical function for any organization that is using the cloud. Azure role-based access control (Azure RBAC) helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to.

Azure RBAC is an authorization system built on [Azure Resource Manager](#) that provides fine-grained access management to Azure resources.

This video provides a quick overview of Azure RBAC.

## What can I do with Azure RBAC?

Here are some examples of what you can do with Azure RBAC:

- Allow one user to manage virtual machines in a subscription and another user to manage virtual networks
- Allow a DBA group to manage SQL databases in a subscription
- Allow a user to manage all resources in a resource group, such as virtual machines, websites, and subnets
- Allow an application to access all resources in a resource group

## How Azure RBAC works

The way you control access to resources using Azure RBAC is to assign Azure roles. This is a key concept to understand – it's how permissions are enforced. A role assignment consists of three elements: security principal, role definition, and scope.

### Security principal

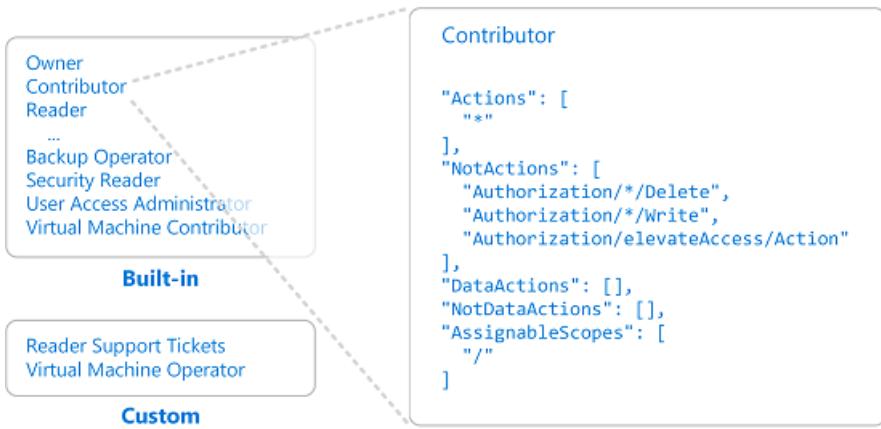
A *security principal* is an object that represents a user, group, service principal, or managed identity that is requesting access to Azure resources. You can assign a role to any of these security principals.



### Role definition

A *role definition* is a collection of permissions. It's typically just called a *role*. A role definition lists the actions that can be performed, such as read, write, and delete. Roles can be high-level, like owner, or specific, like virtual machine reader.

## 2 Role definition



Azure includes several [built-in roles](#) that you can use. For example, the [Virtual Machine Contributor](#) role allows a user to create and manage virtual machines. If the built-in roles don't meet the specific needs of your organization, you can create your own [Azure custom roles](#).

This video provides a quick overview of built-in roles and custom roles.

Azure has data actions that enable you to grant access to data within an object. For example, if a user has read data access to a storage account, then they can read the blobs or messages within that storage account.

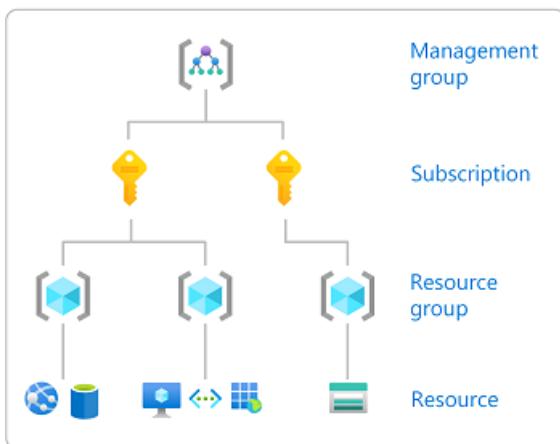
For more information, see [Understand Azure role definitions](#).

## Scope

**Scope** is the set of resources that the access applies to. When you assign a role, you can further limit the actions allowed by defining a scope. This is helpful if you want to make someone a [Website Contributor](#), but only for one resource group.

In Azure, you can specify a scope at four levels: [management group](#), subscription, [resource group](#), or [resource](#). Scopes are structured in a parent-child relationship. You can assign roles at any of these levels of scope.

## 3 Scope



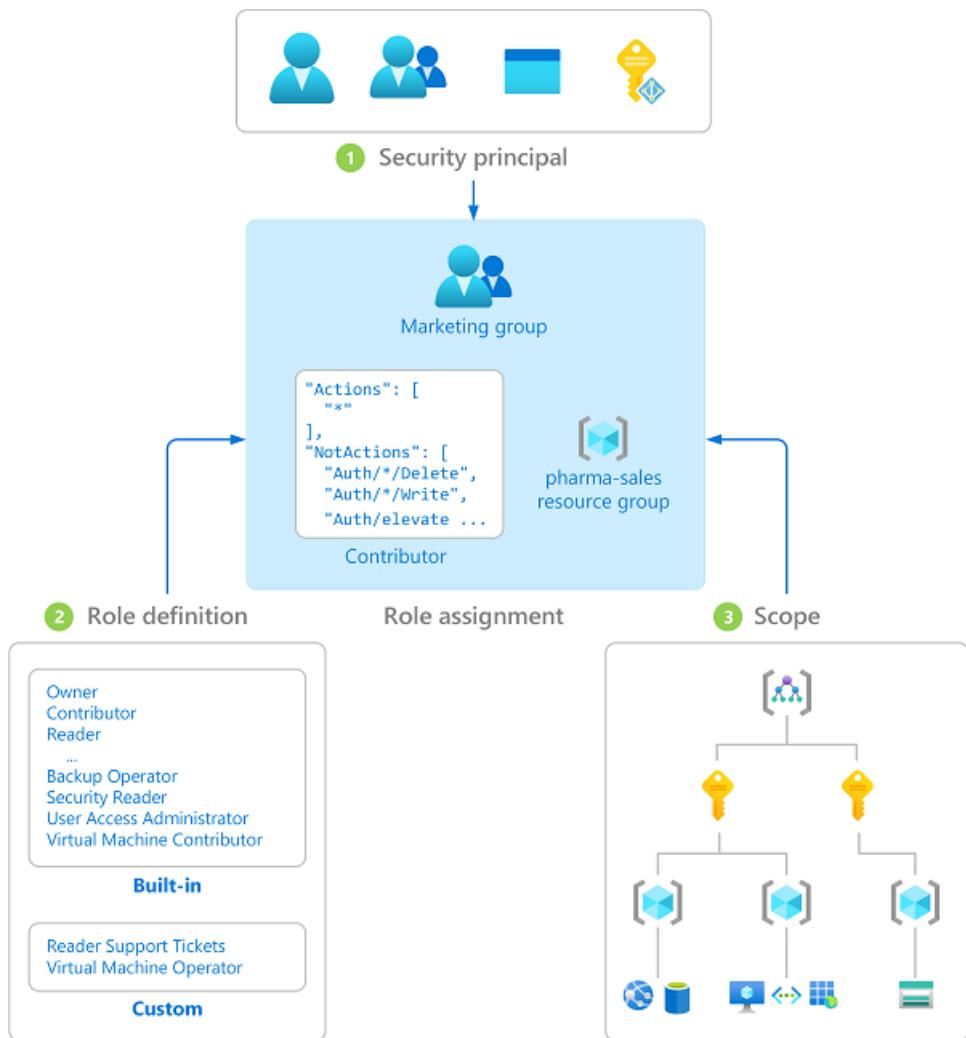
For more information about scope, see [Understand scope](#).

## Role assignments

A *role assignment* is the process of attaching a role definition to a user, group, service principal, or managed identity at a particular scope for the purpose of granting access. Access is granted by creating a role assignment, and access is revoked by removing a role assignment.

The following diagram shows an example of a role assignment. In this example, the Marketing group has been assigned the [Contributor](#) role for the pharma-sales resource group. This means that users in the Marketing

group can create or manage any Azure resource in the pharma-sales resource group. Marketing users do not have access to resources outside the pharma-sales resource group, unless they are part of another role assignment.

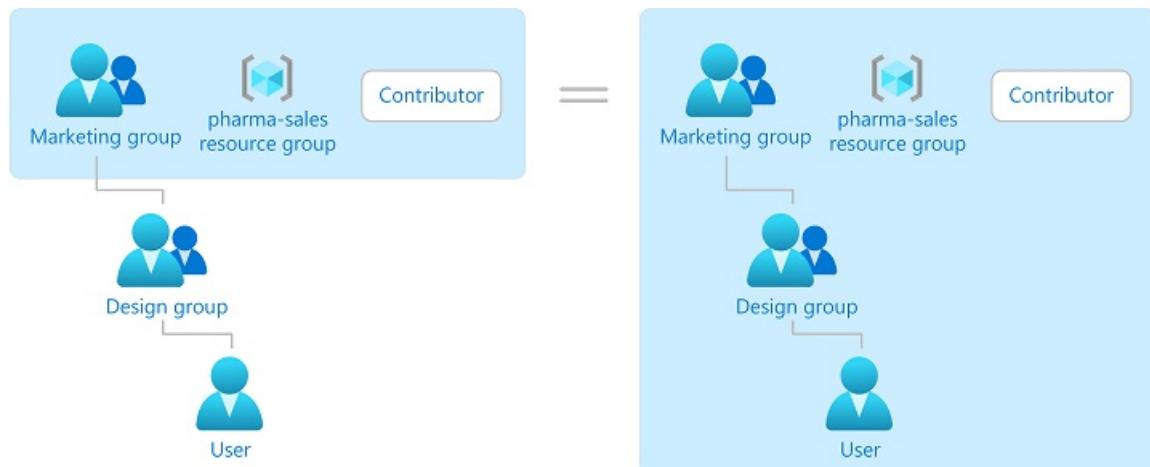


You can assign roles using the Azure portal, Azure CLI, Azure PowerShell, Azure SDKs, or REST APIs.

For more information, see [Steps to assign an Azure role](#).

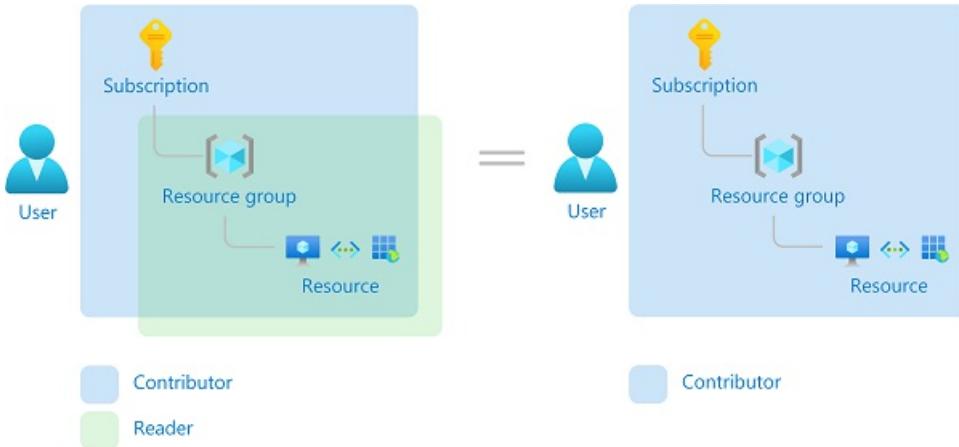
## Groups

Role assignments are transitive for groups which means that if a user is a member of a group and that group is a member of another group that has a role assignment, the user will have the permissions in the role assignment.



## Multiple role assignments

So what happens if you have multiple overlapping role assignments? Azure RBAC is an additive model, so your effective permissions are the sum of your role assignments. Consider the following example where a user is granted the Contributor role at the subscription scope and the Reader role on a resource group. The sum of the Contributor permissions and the Reader permissions is effectively the Contributor role for the subscription. Therefore, in this case, the Reader role assignment has no impact.



## Deny assignments

Previously, Azure RBAC was an allow-only model with no deny, but now Azure RBAC supports deny assignments in a limited way. Similar to a role assignment, a *deny assignment* attaches a set of deny actions to a user, group, service principal, or managed identity at a particular scope for the purpose of denying access. A role assignment defines a set of actions that are *allowed*, while a deny assignment defines a set of actions that are *not allowed*. In other words, deny assignments block users from performing specified actions even if a role assignment grants them access. Deny assignments take precedence over role assignments.

For more information, see [Understand Azure deny assignments](#).

## How Azure RBAC determines if a user has access to a resource

The following are the high-level steps that Azure RBAC uses to determine if you have access to a resource. These steps apply to Azure Resource Manager or data plane services integrated with Azure RBAC. This is helpful to understand if you are trying to troubleshoot an access issue.

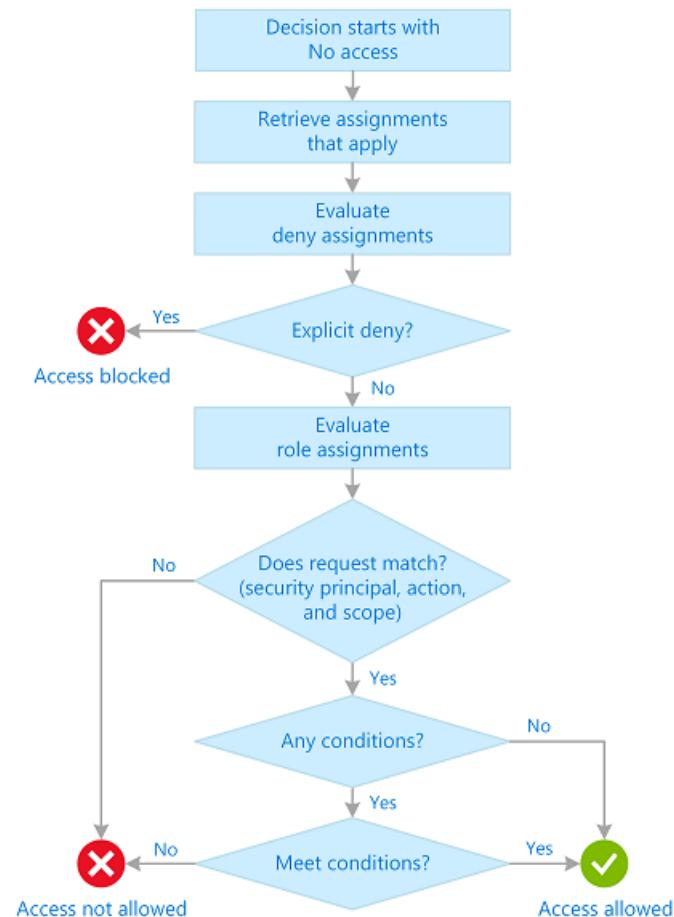
1. A user (or service principal) acquires a token for Azure Resource Manager.  
The token includes the user's group memberships (including transitive group memberships).
2. The user makes a REST API call to Azure Resource Manager with the token attached.
3. Azure Resource Manager retrieves all the role assignments and deny assignments that apply to the resource upon which the action is being taken.
4. If a deny assignment applies, access is blocked. Otherwise, evaluation continues.
5. Azure Resource Manager narrows the role assignments that apply to this user or their group and determines what roles the user has for this resource.
6. Azure Resource Manager determines if the action in the API call is included in the roles the user has for this resource. If the roles include `Actions` that have a wildcard (\*), the effective permissions are computed by subtracting the `NotActions` from the allowed `Actions`. Similarly, the same subtraction is done for any data actions.

Actions - NotActions = Effective management permissions

DataActions - NotDataActions = Effective data permissions

7. If the user doesn't have a role with the action at the requested scope, access is not allowed. Otherwise, any conditions are evaluated.
8. If the role assignment includes conditions, they are evaluated. Otherwise access is allowed.
9. If conditions are met, access is allowed. Otherwise access is not allowed.

The following diagram is a summary of the evaluation logic.



## Where is Azure RBAC data stored?

Role definitions, role assignments, and deny assignments are stored globally to ensure that you have access to your resources regardless of the region you created the resource.

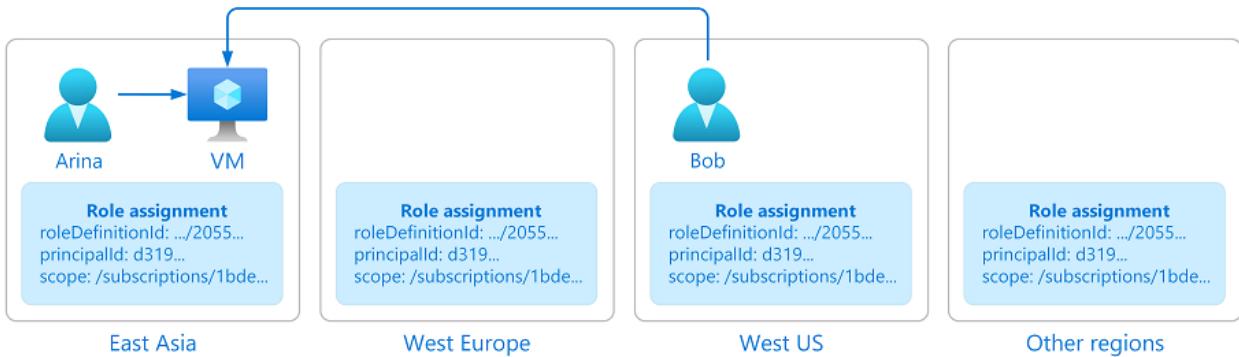
When a role assignment or any other Azure RBAC data is deleted, the data is globally deleted. Principals that had access to a resource via Azure RBAC data will lose their access.

## Why is Azure RBAC data global?

Azure RBAC data is global to ensure that customers can timely access resources regardless from where they are accessing. Azure RBAC is enforced by Azure Resource Manager, which has a global endpoint and requests are routed to the nearest region for speed and resilience. Therefore, Azure RBAC must be enforced in all regions and the data is replicated to all regions. For more information, see [Resiliency of Azure Resource Manager](#).

Consider the following example. Arina creates a virtual machine in East Asia. Bob, who is a member of Arina's team, works in the United States. Bob needs to access the virtual machine that was created in East Asia. To grant Bob timely access to the virtual machine, Azure needs to globally replicate the role assignment that grants Bob

access to the virtual machine from anywhere Bob is.



## License requirements

Using this feature is free and included in your Azure subscription.

## Next steps

- [Assign Azure roles using the Azure portal](#)
- [Understand the different roles](#)
- [Cloud Adoption Framework: Resource access management in Azure](#)

# How to set up Key Vault for virtual machines with the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

In the Azure Resource Manager stack, secrets/certificates are modeled as resources that are provided by Key Vault. To learn more about Azure Key Vault, see [What is Azure Key Vault?](#) In order for Key Vault to be used with Azure Resource Manager VMs, the *EnabledForDeployment* property on Key Vault must be set to true. This article shows you how to set up Key Vault for use with Azure virtual machines (VMs) using the Azure CLI.

To perform these steps, you need the latest [Azure CLI](#) installed and logged in to an Azure account using [az login](#).

## Create a Key Vault

Create a key vault and assign the deployment policy with [az keyvault create](#). The following example creates a key vault named `myKeyVault` in the `myResourceGroup` resource group:

```
az keyvault create -l westus -n myKeyVault -g myResourceGroup --enabled-for-deployment true
```

## Update a Key Vault for use with VMs

Set the deployment policy on an existing key vault with [az keyvault update](#). The following updates the key vault named `myKeyVault` in the `myResourceGroup` resource group:

```
az keyvault update -n myKeyVault -g myResourceGroup --set properties.enabledForDeployment=true
```

## Use templates to set up Key Vault

When you use a template, you need to set the `enabledForDeployment` property to `true` for the Key Vault resource as follows:

```
{
  "type": "Microsoft.KeyVault/vaults",
  "name": "ContosoKeyVault",
  "apiVersion": "2015-06-01",
  "location": "<location-of-key-vault>",
  "properties": {
    "enabledForDeployment": "true",
    ...
    ...
  }
}
```

## Next steps

For other options that you can configure when you create a Key Vault by using templates, see [Create a key vault](#).

# Set up Key Vault for virtual machines using Azure PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

## NOTE

Azure has two different deployment models you can use to create and work with resources: [Azure Resource Manager](#) and [classic](#). This article covers the use of the Resource Manager deployment model. We recommend the Resource Manager deployment model for new deployments instead of the classic deployment model.

In Azure Resource Manager stack, secrets/certificates are modeled as resources that are provided by the resource provider of Key Vault. To learn more about Key Vault, see [What is Azure Key Vault?](#)

## NOTE

1. In order for Key Vault to be used with Azure Resource Manager virtual machines, the `EnabledForDeployment` property on Key Vault must be set to true. You can do this in various clients.
2. The Key Vault needs to be created in the same subscription and location as the Virtual Machine.

## Use PowerShell to set up Key Vault

To create a key vault by using PowerShell, see [Set and retrieve a secret from Azure Key Vault using PowerShell](#).

For new key vaults, you can use this PowerShell cmdlet:

```
New-AzKeyVault -VaultName 'ContosoKeyVault' -ResourceGroupName 'ContosoResourceGroup' -Location 'East Asia'  
-EnabledForDeployment
```

For existing key vaults, you can use this PowerShell cmdlet:

```
Set-AzKeyVaultAccessPolicy -VaultName 'ContosoKeyVault' -EnabledForDeployment
```

## Use CLI to set up Key Vault

To create a key vault by using the command-line interface (CLI), see [Manage Key Vault using CLI](#).

For CLI, you have to create the key vault before you assign the deployment policy. You can do this by using the following command:

```
az keyvault create --name "ContosoKeyVault" --resource-group "ContosoResourceGroup" --location "EastAsia"
```

Then to enable Key Vault for use with template deployment, run the following command:

```
az keyvault update --name "ContosoKeyVault" --resource-group "ContosoResourceGroup" --enabled-for-deployment  
"true"
```

## Use templates to set up Key Vault

While you use a template, you need to set the `enabledForDeployment` property to `true` for the Key Vault resource.

```
{  
  "type": "Microsoft.KeyVault/vaults",  
  "name": "ContosoKeyVault",  
  "apiVersion": "2015-06-01",  
  "location": "<location-of-key-vault>",  
  "properties": {  
    "enabledForDeployment": "true",  
    ....  
    ....  
  }  
}
```

For other options that you can configure when you create a key vault by using templates, see [Create a key vault](#).

# Guidance for mitigating silicon based micro-architectural and speculative execution side-channel vulnerabilities

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article provides guidance for a new class of silicon based micro-architectural and speculative execution side-channel vulnerabilities that affect many modern processors and operating systems. This includes Intel, AMD, and ARM. Specific details for these silicon-based vulnerabilities can be found in the following security advisories and CVEs:

- [ADV180002 - Guidance to mitigate speculative execution side-channel vulnerabilities](#)
- [ADV180012 - Microsoft Guidance for Speculative Store Bypass](#)
- [ADV180013 - Microsoft Guidance for Rogue System Register Read](#)
- [ADV180016 - Microsoft Guidance for Lazy FP State Restore](#)
- [ADV180018 - Microsoft Guidance to mitigate L1TF variant](#)
- [ADV190013 - Microsoft Guidance to mitigate Microarchitectural Data Sampling vulnerabilities](#)
- [ADV220002 - Microsoft Guidance on Intel Processor MMIO Stale Data Vulnerabilities](#)
- [CVE-2022-23816](<https://msrc.microsoft.com/update-guide/vulnerability/CVE-2022-23816> - AMD CPU Branch Type Confusion)
- [CVE-2022-21123](<https://msrc.microsoft.com/update-guide/vulnerability/CVE-2022-23825> - AMD CPU Branch Type Confusion)

The disclosure of these CPU vulnerabilities has resulted in questions from customers seeking more clarity.

Microsoft has deployed mitigations across all our cloud services. The infrastructure that runs Azure and isolates customer workloads from each other is protected. This means that a potential attacker using the same infrastructure can't attack your application using these vulnerabilities.

Azure is using [memory preserving maintenance](#) whenever possible, to minimize customer impact and eliminate the need for reboots. Azure will continue utilizing these methods when making systemwide updates to the host and protect our customers.

More information about how security is integrated into every aspect of Azure is available on the [Azure Security Documentation](#) site.

## NOTE

Since this document was first published, multiple variants of this vulnerability class have been disclosed. Microsoft continues to be heavily invested in protecting our customers and providing guidance. This page will be updated as we continue to release further fixes.

**Customers that are running untrusted code within their VM** need to take action to protect against these vulnerabilities by reading below for more guidance on all vulnerabilities.

Other customers should evaluate these vulnerabilities from a Defense in Depth perspective and consider the security and performance implications of their chosen configuration.

# Keeping your operating systems up-to-date

While an OS update is not required to isolate your applications running on Azure from other Azure customers, it is always a best practice to keep your software up-to-date. The latest Security Updates for Windows contain mitigations for these vulnerabilities. Similarly, Linux distributions have released multiple updates to address these vulnerabilities. Here are our recommended actions to update your operating system:

OFFERING	RECOMMENDED ACTION
Azure Cloud Services	Enable <a href="#">auto update</a> or ensure you're running the newest Guest OS.
Azure Linux Virtual Machines	Install updates from your operating system provider. For more information, see <a href="#">Linux</a> later in this document.
Azure Windows Virtual Machines	Install the latest security rollup.
Other Azure PaaS Services	There is no action needed for customers using these services. Azure automatically keeps your OS versions up-to-date.

## Additional guidance if you're running untrusted code

Customers who allow untrusted users to execute arbitrary code may wish to implement some extra security features inside their Azure Virtual Machines or Cloud Services. These features protect against the intra-process disclosure vectors that several speculative execution vulnerabilities describe.

Example scenarios where more security features are recommended:

- You allow code that you do not trust to run inside your VM.
  - *For example, you allow one of your customers to upload a binary or script that you then execute within your application.*
- You allow users that you do not trust to log into your VM using low privileged accounts.
  - *For example, you allow a low-privileged user to log into one of your VMs using remote desktop or SSH.*
- You allow untrusted users access to virtual machines implemented via nested virtualization.
  - *For example, you control the Hyper-V host, but allocate the VMs to untrusted users.*

Customers who do not implement a scenario involving untrusted code do not need to enable these extra security features.

## Enabling additional security

You can enable more security features inside your VM or Cloud Service if you're running untrusted code. In parallel, ensure your operating system is up-to-date to enable security features inside your VM or Cloud Service

### Windows

Your target operating system must be up-to-date to enable these extra security features. While numerous mitigations are enabled by default, the extra features described here must be enabled manually and may cause a performance impact.

#### Option 1

Step 1: Follow the instructions in [KB4072698](#) to verify protections are enabled using the [SpeculationControl](#) PowerShell module.

#### NOTE

If you previously downloaded this module, you will need to install the newest version.

To validate enabled protections against these vulnerabilities, see [Understanding Get-SpeculationControlSettings PowerShell script output](#).

If protections are not enabled, please [contact Azure Support](#) to enable additional controls on your Azure VM.

**Step 2:** To enable Kernel Virtual Address Shadowing (KVAS) and Branch Target Injection (BTI) OS support, follow the instructions in [KB4072698](#) to enable protections using the `Session Manager` registry keys. A reboot is required.

**Step 3:** For deployments that are using [nested virtualization](#) (D3 and E3 only): These instructions apply inside the VM you're using as a Hyper-V host.

1. Follow the instructions in [KB4072698](#) to enable protections using the `MinVmVersionForCpuBasedMitigations` registry keys.
2. Set the hypervisor scheduler type to `Core` by following the instructions [here](#).

#### Option 2

**Disable hyper-threading on the VM** - Customers running untrusted code on a hyper-threaded VM might choose to disable hyper-threading or move to a non-hyper-threaded VM size. Reference [this doc](#) for a list of hyper-threaded VM sizes (where ratio of vCPU to Core is 2:1). To check if your VM has hyper-threading enabled, refer to the below script using the Windows command line from within the VM.

Type `wmic` to enter the interactive interface. Then type the following command to view the amount of physical and logical processors on the VM.

```
CPU Get NumberOfCores,NumberOfLogicalProcessors /Format>List
```

If the number of logical processors is greater than physical processors (cores), then hyper-threading is enabled. If you're running a hyper-threaded VM, [contact Azure Support](#) to get hyper-threading disabled. Once hyper-threading is disabled, support will require a full VM reboot. Refer to [Core count](#) to understand why your VM core count decreased.

#### Option 3

For [CVE-2022-23816](#) and [CVE-2022-21123](#) (AMD CPU Branch Type Confusion), follow both **Option 1** and **Option 2** above.

#### Linux

Enabling the set of extra security features inside requires that the target operating system be fully up-to-date. Some mitigations will be enabled by default. The following section describes the features which are off by default and/or reliant on hardware support (microcode). Enabling these features may cause a performance impact. Reference your operating system provider's documentation for further instructions

**Step 1: Disable hyper-threading on the VM** - Customers running untrusted code on a hyper-threaded VM will need to disable hyper-threading or move to a non-hyper-threaded VM. Reference [this doc](#) for a list of hyper-threaded VM sizes (where ratio of vCPU to Core is 2:1). To check if you're running a hyper-threaded VM, run the `lscpu` command in the Linux VM.

If `Thread(s) per core = 2`, then hyper-threading has been enabled.

If `Thread(s) per core = 1`, then hyper-threading has been disabled.

Sample output for a VM with hyper-threading enabled:

```
CPU Architecture:      x86_64
CPU op-mode(s):       32-bit, 64-bit
Byte Order:           Little Endian
CPU(s):               8
On-line CPU(s) list: 0-7
Thread(s) per core:   2
Core(s) per socket:   4
Socket(s):            1
NUMA node(s):         1
```

If you're running a hyper-threaded VM, [contact Azure Support](#) to get hyper-threading disabled. Once hyper-threading is disabled, **support will require a full VM reboot**. Refer to [Core count](#) to understand why your VM core count decreased.

**Step 2:** To mitigate against any of the below CPU based memory vulnerabilities, refer to your operating system provider's documentation:

- [Redhat and CentOS](#)
- [SUSE](#)
- [Ubuntu](#)

#### Core count

When a hyper-threaded VM is created, Azure allocates 2 threads per core - these are called vCPUs. When hyper-threading is disabled, Azure removes a thread and surfaces up single threaded cores (physical cores). The ratio of vCPU to CPU is 2:1, so once hyper-threading is disabled, the CPU count in the VM will appear to have decreased by half. For example, a D8\_v3 VM is a hyper-threaded VM running on 8 vCPUs (2 threads per core x 4 cores). When hyper-threading is disabled, CPUs will drop to 4 physical cores with 1 thread per core.

## Next steps

For more information about how security is integrated into every aspect of Azure, see [Azure Security Documentation](#).

# Join a Red Hat Enterprise Linux virtual machine to an Azure Active Directory Domain Services managed domain

9/21/2022 • 8 minutes to read • [Edit Online](#)

To let users sign in to virtual machines (VMs) in Azure using a single set of credentials, you can join VMs to an Azure Active Directory Domain Services (Azure AD DS) managed domain. When you join a VM to an Azure AD DS managed domain, user accounts and credentials from the domain can be used to sign in and manage servers. Group memberships from the managed domain are also applied to let you control access to files or services on the VM.

This article shows you how to join a Red Hat Enterprise Linux (RHEL) VM to a managed domain.

## Prerequisites

To complete this tutorial, you need the following resources and privileges:

- An active Azure subscription.
  - If you don't have an Azure subscription, [create an account](#).
- An Azure Active Directory tenant associated with your subscription, either synchronized with an on-premises directory or a cloud-only directory.
  - If needed, [create an Azure Active Directory tenant](#) or [associate an Azure subscription with your account](#).
- An Azure Active Directory Domain Services managed domain enabled and configured in your Azure AD tenant.
  - If needed, the first tutorial [creates and configures an Azure Active Directory Domain Services managed domain](#).
- A user account that's a part of the managed domain.
- Unique Linux VM names that are a maximum of 15 characters to avoid truncated names that might cause conflicts in Active Directory.

## Create and connect to a RHEL Linux VM

If you have an existing RHEL Linux VM in Azure, connect to it using SSH, then continue on to the next step to [start configuring the VM](#).

If you need to create a RHEL Linux VM, or want to create a test VM for use with this article, you can use one of the following methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Azure PowerShell](#)

When you create the VM, pay attention to the virtual network settings to make sure that the VM can communicate with the managed domain:

- Deploy the VM into the same, or a peered, virtual network in which you have enabled Azure AD Domain Services.
- Deploy the VM into a different subnet than your Azure AD Domain Services managed domain.

Once the VM is deployed, follow the steps to connect to the VM using SSH.

## Configure the hosts file

To make sure that the VM host name is correctly configured for the managed domain, edit the `/etc/hosts` file and set the hostname:

```
sudo vi /etc/hosts
```

In the `hosts` file, update the `localhost` address. In the following example:

- `aaddscontoso.com` is the DNS domain name of your managed domain.
- `rhel` is the hostname of your RHEL VM that you're joining to the managed domain.

Update these names with your own values:

```
127.0.0.1 rhel rhel.aaddscontoso.com
```

When done, save and exit the `hosts` file using the `:wq` command of the editor.

## Install required packages

The VM needs some additional packages to join the VM to the managed domain. To install and configure these packages, update and install the domain-join tools using `yum`. There are some differences between RHEL 7.x and RHEL 6.x, so use the appropriate commands for your distro version in the remaining sections of this article.

### RHEL 7

```
sudo yum install realmd sssd krb5-workstation krb5-libs oddjob oddjob-mkhomedir samba-common-tools
```

### RHEL 6

```
sudo yum install adcli sssd authconfig krb5-workstation
```

## Join VM to the managed domain

Now that the required packages are installed on the VM, join the VM to the managed domain. Again, use the appropriate steps for your RHEL distro version.

### RHEL 7

1. Use the `realm discover` command to discover the managed domain. The following example discovers the realm `AADDSCONTOSO.COM`. Specify your own managed domain name in ALL UPPERCASE:

```
sudo realm discover AADDSCONTOSO.COM
```

If the `realm discover` command can't find your managed domain, review the following troubleshooting steps:

- Make sure that the domain is reachable from the VM. Try `ping aaddscontoso.com` to see if a positive reply is returned.
- Check that the VM is deployed to the same, or a peered, virtual network in which the managed

domain is available.

- Confirm that the DNS server settings for the virtual network have been updated to point to the domain controllers of the managed domain.
2. Now initialize Kerberos using the `kinit` command. Specify a user that's a part of the managed domain. If needed, [add a user account to a group in Azure AD](#).

Again, the managed domain name must be entered in ALL UPPERCASE. In the following example, the account named `contosoadmin@aaddscontoso.com` is used to initialize Kerberos. Enter your own user account that's a part of the managed domain:

```
kinit contosoadmin@AADDSCONTOSO.COM
```

3. Finally, join the VM to the managed domain using the `realm join` command. Use the same user account that's a part of the managed domain that you specified in the previous `kinit` command, such as `contosoadmin@AADDSCONTOSO.COM`:

```
sudo realm join --verbose AADDSCONTOSO.COM -U 'contosoadmin@AADDSCONTOSO.COM'
```

It takes a few moments to join the VM to the managed domain. The following example output shows the VM has successfully joined to the managed domain:

```
Successfully enrolled machine in realm
```

## RHEL 6

1. Use the `adcli info` command to discover the managed domain. The following example discovers the realm `AADDSCONTOSO.COM`. Specify your own managed domain name in ALL UPPERCASE:

```
sudo adcli info aaddscontoso.com
```

If the `adcli info` command can't find your managed domain, review the following troubleshooting steps:

- Make sure that the domain is reachable from the VM. Try `ping aaddscontoso.com` to see if a positive reply is returned.
  - Check that the VM is deployed to the same, or a peered, virtual network in which the managed domain is available.
  - Confirm that the DNS server settings for the virtual network have been updated to point to the domain controllers of the managed domain.
2. First, join the domain using the `adcli join` command, this command also creates the keytab to authenticate the machine. Use a user account that's a part of the managed domain.

```
sudo adcli join aaddscontoso.com -U contosoadmin
```

3. Now configure the `/etc/krb5.conf` and create the `/etc/sssd/sssd.conf` files to use the `aaddscontoso.com` Active Directory domain. Make sure that `AADDSCONTOSO.COM` is replaced by your own domain name:

Open the `/etc/krb5.conf` file with an editor:

```
sudo vi /etc/krb5.conf
```

Update the `krb5.conf` file to match the following sample:

```
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
default_realm = AADDSCONTOSO.COM
dns_lookup_realm = true
dns_lookup_kdc = true
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true

[realms]
AADDSCONTOSO.COM = {
kdc = AADDSCONTOSO.COM
admin_server = AADDSCONTOSO.COM
}

[domain_realm]
.AADDSCONTOSO.COM = AADDSCONTOSO.COM
AADDSCONTOSO.COM = AADDSCONTOSO.COM
```

Create the `/etc/sssd/sssd.conf` file:

```
sudo vi /etc/sssd/sssd.conf
```

Update the `sssd.conf` file to match the following sample:

```
[sssd]
services = nss, pam, ssh, autofs
config_file_version = 2
domains = AADDSCONTOSO.COM

[domain/AADDSCONTOSO.COM]

id_provider = ad
```

4. Make sure `/etc/sssd/sssd.conf` permissions are 600 and is owned by root user:

```
sudo chmod 600 /etc/sssd/sssd.conf
sudo chown root:root /etc/sssd/sssd.conf
```

5. Use `authconfig` to instruct the VM about the AD Linux integration:

```
sudo authconfig --enablesssd --enablesssdauth --update
```

6. Start and enable the sssd service:

```
sudo service sssd start
sudo chkconfig sssd on
```

If your VM can't successfully complete the domain-join process, make sure that the VM's network security group allows outbound Kerberos traffic on TCP + UDP port 464 to the virtual network subnet for your managed

domain.

Now check if you can query user AD information using `getent`

```
sudo getent passwd contosoadmin
```

## Allow password authentication for SSH

By default, users can only sign in to a VM using SSH public key-based authentication. Password-based authentication fails. When you join the VM to a managed domain, those domain accounts need to use password-based authentication. Update the SSH configuration to allow password-based authentication as follows.

1. Open the `sshd_config` file with an editor:

```
sudo vi /etc/ssh/sshd_config
```

2. Update the line for `PasswordAuthentication` to `yes`:

```
PasswordAuthentication yes
```

When done, save and exit the `sshd_config` file using the `:wq` command of the editor.

3. To apply the changes and let users sign in using a password, restart the SSH service for your RHEL distro version:

RHEL 7

```
sudo systemctl restart sshd
```

RHEL 6

```
sudo service sshd restart
```

## Grant the 'AAD DC Administrators' group sudo privileges

To grant members of the `AAD DC Administrators` group administrative privileges on the RHEL VM, you add an entry to the `/etc/sudoers`. Once added, members of the `AAD DC Administrators` group can use the `sudo` command on the RHEL VM.

1. Open the `sudoers` file for editing:

```
sudo visudo
```

2. Add the following entry to the end of `/etc/sudoers` file. The `AAD DC Administrators` group contains whitespace in the name, so include the backslash escape character in the group name. Add your own domain name, such as `aaddscontoso.com`:

```
# Add 'AAD DC Administrators' group members as admins.  
%AAD\ DC\ Administrators@aaddscontoso.com ALL=(ALL) NOPASSWD:ALL
```

When done, save and exit the editor using the `:wq` command of the editor.

## Sign in to the VM using a domain account

To verify that the VM has been successfully joined to the managed domain, start a new SSH connection using a domain user account. Confirm that a home directory has been created, and that group membership from the domain is applied.

1. Create a new SSH connection from your console. Use a domain account that belongs to the managed domain using the `ssh -l` command, such as `contosoadmin@aaddscontoso.com` and then enter the address of your VM, such as `rhel.aaddscontoso.com`. If you use the Azure Cloud Shell, use the public IP address of the VM rather than the internal DNS name.

```
ssh -l contosoadmin@AADDSCONTOSO.com rhel.aaddscontoso.com
```

2. When you've successfully connected to the VM, verify that the home directory was initialized correctly:

```
pwd
```

You should be in the `/home` directory with your own directory that matches the user account.

3. Now check that the group memberships are being resolved correctly:

```
id
```

You should see your group memberships from the managed domain.

4. If you signed in to the VM as a member of the *AAD DC Administrators* group, check that you can correctly use the `sudo` command:

```
sudo yum update
```

## Next steps

If you have problems connecting the VM to the managed domain or signing in with a domain account, see [Troubleshooting domain join issues](#).

# Join a CentOS Linux virtual machine to an Azure Active Directory Domain Services managed domain

9/21/2022 • 6 minutes to read • [Edit Online](#)

To let users sign in to virtual machines (VMs) in Azure using a single set of credentials, you can join VMs to an Azure Active Directory Domain Services (Azure AD DS) managed domain. When you join a VM to an Azure AD DS managed domain, user accounts and credentials from the domain can be used to sign in and manage servers. Group memberships from the managed domain are also applied to let you control access to files or services on the VM.

This article shows you how to join a CentOS Linux VM to a managed domain.

## Prerequisites

To complete this tutorial, you need the following resources and privileges:

- An active Azure subscription.
  - If you don't have an Azure subscription, [create an account](#).
- An Azure Active Directory tenant associated with your subscription, either synchronized with an on-premises directory or a cloud-only directory.
  - If needed, [create an Azure Active Directory tenant](#) or [associate an Azure subscription with your account](#).
- An Azure Active Directory Domain Services managed domain enabled and configured in your Azure AD tenant.
  - If needed, the first tutorial [creates and configures an Azure Active Directory Domain Services managed domain](#).
- A user account that's part of the managed domain.
- Unique Linux VM names that are a maximum of 15 characters to avoid truncated names that might cause conflicts in Active Directory.

## Create and connect to a CentOS Linux VM

If you have an existing CentOS Linux VM in Azure, connect to it using SSH, then continue on to the next step to [start configuring the VM](#).

If you need to create a CentOS Linux VM, or want to create a test VM for use with this article, you can use one of the following methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Azure PowerShell](#)

When you create the VM, pay attention to the virtual network settings to make sure that the VM can communicate with the managed domain:

- Deploy the VM into the same, or a peered, virtual network in which you have enabled Azure AD Domain Services.
- Deploy the VM into a different subnet than your managed domain.

Once the VM is deployed, follow the steps to connect to the VM using SSH.

# Configure the hosts file

To make sure that the VM host name is correctly configured for the managed domain, edit the `/etc/hosts` file and set the hostname:

```
sudo vi /etc/hosts
```

In the `hosts` file, update the `localhost` address. In the following example:

- `aaddscontoso.com` is the DNS domain name of your managed domain.
- `centos` is the hostname of your CentOS VM that you're joining to the managed domain.

Update these names with your own values:

```
127.0.0.1 centos.aaddscontoso.com centos
```

When done, save and exit the `hosts` file using the `:wq` command of the editor.

## Install required packages

The VM needs some additional packages to join the VM to the managed domain. To install and configure these packages, update and install the domain-join tools using `yum`:

```
sudo yum install adcli realmd sssd krb5-workstation krb5-libs oddjob oddjob-mkhomedir samba-common-tools
```

## Join VM to the managed domain

Now that the required packages are installed on the VM, join the VM to the managed domain.

1. Use the `realm discover` command to discover the managed domain. The following example discovers the realm `AADDSCONTOSO.COM`. Specify your own managed domain name in ALL UPPERCASE:

```
sudo realm discover AADDSCONTOSO.COM
```

If the `realm discover` command can't find your managed domain, review the following troubleshooting steps:

- Make sure that the domain is reachable from the VM. Try `ping aaddscontoso.com` to see if a positive reply is returned.
  - Check that the VM is deployed to the same, or a peered, virtual network in which the managed domain is available.
  - Confirm that the DNS server settings for the virtual network have been updated to point to the domain controllers of the managed domain.
2. Now initialize Kerberos using the `kinit` command. Specify a user that's a part of the managed domain. If needed, [add a user account to a group in Azure AD](#).

Again, the managed domain name must be entered in ALL UPPERCASE. In the following example, the account named `contosoadmin@aaddscontoso.com` is used to initialize Kerberos. Enter your own user account that's a part of the managed domain:

```
kinit contosoadmin@AADDSCONTOSO.COM
```

3. Finally, join the VM to the managed domain using the `realm join` command. Use the same user account that's a part of the managed domain that you specified in the previous `kinit` command, such as

```
contosoadmin@AADDSCONTOSO.COM :
```

```
sudo realm join --verbose AADDSCONTOSO.COM -U 'contosoadmin@AADDSCONTOSO.COM' --membership-software=adcli
```

It takes a few moments to join the VM to the managed domain. The following example output shows the VM has successfully joined to the managed domain:

```
Successfully enrolled machine in realm
```

If your VM can't successfully complete the domain-join process, make sure that the VM's network security group allows outbound Kerberos traffic on TCP + UDP port 464 to the virtual network subnet for your managed domain.

## Allow password authentication for SSH

By default, users can only sign in to a VM using SSH public key-based authentication. Password-based authentication fails. When you join the VM to a managed domain, those domain accounts need to use password-based authentication. Update the SSH configuration to allow password-based authentication as follows.

1. Open the `sshd_config` file with an editor:

```
sudo vi /etc/ssh/sshd_config
```

2. Update the line for `PasswordAuthentication` to `yes`:

```
PasswordAuthentication yes
```

When done, save and exit the `sshd_config` file using the `:wq` command of the editor.

3. To apply the changes and let users sign in using a password, restart the SSH service:

```
sudo systemctl restart sshd
```

## Grant the 'AAD DC Administrators' group sudo privileges

To grant members of the `AAD DC Administrators` group administrative privileges on the CentOS VM, you add an entry to the `/etc/sudoers`. Once added, members of the `AAD DC Administrators` group can use the `sudo` command on the CentOS VM.

1. Open the `sudoers` file for editing:

```
sudo visudo
```

2. Add the following entry to the end of `/etc/sudoers` file. The `AAD DC Administrators` group contains whitespace in the name, so include the backslash escape character in the group name. Add your own domain name, such as `aaddscontoso.com`:

```
# Add 'AAD DC Administrators' group members as admins.  
%AAD\ DC\ Administrators@aaddscontoso.com ALL=(ALL) NOPASSWD:ALL
```

When done, save and exit the editor using the `:wq` command of the editor.

## Sign in to the VM using a domain account

To verify that the VM has been successfully joined to the managed domain, start a new SSH connection using a domain user account. Confirm that a home directory has been created, and that group membership from the domain is applied.

1. Create a new SSH connection from your console. Use a domain account that belongs to the managed domain using the `ssh -l` command, such as `contosoadmin@aaddscontoso.com` and then enter the address of your VM, such as `centos.aaddscontoso.com`. If you use the Azure Cloud Shell, use the public IP address of the VM rather than the internal DNS name.

```
ssh -l contosoadmin@AADDSCONTOSO.com centos.aaddscontoso.com
```

2. When you've successfully connected to the VM, verify that the home directory was initialized correctly:

```
pwd
```

You should be in the `/home` directory with your own directory that matches the user account.

3. Now check that the group memberships are being resolved correctly:

```
id
```

You should see your group memberships from the managed domain.

4. If you signed in to the VM as a member of the *AAD DC Administrators* group, check that you can correctly use the `sudo` command:

```
sudo yum update
```

## Next steps

If you have problems connecting the VM to the managed domain or signing in with a domain account, see [Troubleshooting domain join issues](#).

# Join an Ubuntu Linux virtual machine to an Azure Active Directory Domain Services managed domain

9/21/2022 • 8 minutes to read • [Edit Online](#)

To let users sign in to virtual machines (VMs) in Azure using a single set of credentials, you can join VMs to an Azure Active Directory Domain Services (Azure AD DS) managed domain. When you join a VM to an Azure AD DS managed domain, user accounts and credentials from the domain can be used to sign in and manage servers. Group memberships from the managed domain are also applied to let you control access to files or services on the VM.

This article shows you how to join an Ubuntu Linux VM to a managed domain.

## Prerequisites

To complete this tutorial, you need the following resources and privileges:

- An active Azure subscription.
  - If you don't have an Azure subscription, [create an account](#).
- An Azure Active Directory tenant associated with your subscription, either synchronized with an on-premises directory or a cloud-only directory.
  - If needed, [create an Azure Active Directory tenant](#) or [associate an Azure subscription with your account](#).
- An Azure Active Directory Domain Services managed domain enabled and configured in your Azure AD tenant.
  - If needed, the first tutorial [creates and configures an Azure Active Directory Domain Services managed domain](#).
- A user account that's a part of the managed domain. Make sure the SAMAccountName attribute for the user is not autogenerated. If multiple user accounts in the Azure AD tenant have the same mailNickname attribute, the SAMAccountName attribute for each user is autogenerated. For more information, see [How objects and credentials are synchronized in an Azure Active Directory Domain Services managed domain](#).
- Unique Linux VM names that are a maximum of 15 characters to avoid truncated names that might cause conflicts in Active Directory.

## Create and connect to an Ubuntu Linux VM

If you have an existing Ubuntu Linux VM in Azure, connect to it using SSH, then continue on to the next step to [start configuring the VM](#).

If you need to create an Ubuntu Linux VM, or want to create a test VM for use with this article, you can use one of the following methods:

- [Azure portal](#)
- [Azure CLI](#)
- [Azure PowerShell](#)

When you create the VM, pay attention to the virtual network settings to make sure that the VM can communicate with the managed domain:

- Deploy the VM into the same, or a peered, virtual network in which you have enabled Azure AD Domain

Services.

- Deploy the VM into a different subnet than your Azure AD Domain Services managed domain.

Once the VM is deployed, follow the steps to connect to the VM using SSH.

## Configure the hosts file

To make sure that the VM host name is correctly configured for the managed domain, edit the `/etc/hosts` file and set the hostname:

```
sudo vi /etc/hosts
```

In the `hosts` file, update the `localhost` address. In the following example:

- `aaddscontoso.com` is the DNS domain name of your managed domain.
- `ubuntu` is the hostname of your Ubuntu VM that you're joining to the managed domain.

Update these names with your own values:

```
127.0.0.1 ubuntu.aaddscontoso.com ubuntu
```

When done, save and exit the `hosts` file using the `:wq` command of the editor.

## Install required packages

The VM needs some additional packages to join the VM to the managed domain. To install and configure these packages, update and install the domain-join tools using `apt-get`

During the Kerberos installation, the `krb5-user` package prompts for the realm name in ALL UPPERCASE. For example, if the name of your managed domain is `aaddscontoso.com`, enter `AADDSCONTOSO.COM` as the realm. The installation writes the `[realm]` and `[domain_realm]` sections in `/etc krb5.conf` configuration file. Make sure that you specify the realm an ALL UPPERCASE:

```
sudo apt-get update
sudo apt-get install krb5-user samba sssd sssd-tools libnss-sss libpam-sss ntp ntpdate realmd adcli
```

## Configure Network Time Protocol (NTP)

For domain communication to work correctly, the date and time of your Ubuntu VM must synchronize with the managed domain. Add your managed domain's NTP hostname to the `/etc/ntp.conf` file.

1. Open the `ntp.conf` file with an editor:

```
sudo vi /etc/ntp.conf
```

2. In the `ntp.conf` file, create a line to add your managed domain's DNS name. In the following example, an entry for `aaddscontoso.com` is added. Use your own DNS name:

```
server aaddscontoso.com
```

When done, save and exit the `ntp.conf` file using the `:wq` command of the editor.

3. To make sure that the VM is synchronized with the managed domain, the following steps are needed:

- Stop the NTP server
- Update the date and time from the managed domain
- Start the NTP service

Run the following commands to complete these steps. Use your own DNS name with the `ntpd` command:

```
sudo systemctl stop ntp
sudo ntpdate aaddscontoso.com
sudo systemctl start ntp
```

## Join VM to the managed domain

Now that the required packages are installed on the VM and NTP is configured, join the VM to the managed domain.

1. Use the `realm discover` command to discover the managed domain. The following example discovers the realm `AADDSCONTOSO.COM`. Specify your own managed domain name in ALL UPPERCASE:

```
sudo realm discover AADDSCONTOSO.COM
```

If the `realm discover` command can't find your managed domain, review the following troubleshooting steps:

- Make sure that the domain is reachable from the VM. Try `ping aaddscontoso.com` to see if a positive reply is returned.
- Check that the VM is deployed to the same, or a peered, virtual network in which the managed domain is available.
- Confirm that the DNS server settings for the virtual network have been updated to point to the domain controllers of the managed domain.

2. Now initialize Kerberos using the `kinit` command. Specify a user that's a part of the managed domain. If needed, [add a user account to a group in Azure AD](#).

Again, the managed domain name must be entered in ALL UPPERCASE. In the following example, the account named `contosoadmin@aaddscontoso.com` is used to initialize Kerberos. Enter your own user account that's a part of the managed domain:

```
kinit -V contosoadmin@AADDSCONTOSO.COM
```

3. Finally, join the VM to the managed domain using the `realm join` command. Use the same user account that's a part of the managed domain that you specified in the previous `kinit` command, such as `contosoadmin@AADDSCONTOSO.COM`:

```
sudo realm join --verbose AADDSCONTOSO.COM -U 'contosoadmin@AADDSCONTOSO.COM' --install=/
```

It takes a few moments to join the VM to the managed domain. The following example output shows the VM has successfully joined to the managed domain:

```
Successfully enrolled machine in realm
```

If your VM can't successfully complete the domain-join process, make sure that the VM's network security group allows outbound Kerberos traffic on TCP + UDP port 464 to the virtual network subnet for your managed domain.

If you received the error *Unspecified GSS failure. Minor code may provide more information (Server not found in Kerberos database)*, open the file `/etc/krb5.conf` and add the following code in `[libdefaults]` section and try again:

```
rdns=false
```

## Update the SSSD configuration

One of the packages installed in a previous step was for System Security Services Daemon (SSSD). When a user tries to sign in to a VM using domain credentials, SSSD relays the request to an authentication provider. In this scenario, SSSD uses Azure AD DS to authenticate the request.

1. Open the `sssd.conf` file with an editor:

```
sudo vi /etc/sssd/sssd.conf
```

2. Comment out the line for `use_fully_qualified_names` as follows:

```
# use_fully_qualified_names = True
```

When done, save and exit the `sssd.conf` file using the `:wq` command of the editor.

3. To apply the change, restart the SSSD service:

```
sudo systemctl restart sssd
```

## Configure user account and group settings

With the VM joined to the managed domain and configured for authentication, there are a few user configuration options to complete. These configuration changes include allowing password-based authentication, and automatically creating home directories on the local VM when domain users first sign in.

### Allow password authentication for SSH

By default, users can only sign in to a VM using SSH public key-based authentication. Password-based authentication fails. When you join the VM to a managed domain, those domain accounts need to use password-based authentication. Update the SSH configuration to allow password-based authentication as follows.

1. Open the `sshd_config` file with an editor:

```
sudo vi /etc/ssh/sshd_config
```

2. Update the line for `PasswordAuthentication` to `yes`:

```
PasswordAuthentication yes
```

When done, save and exit the `sshd_config` file using the `:wq` command of the editor.

- To apply the changes and let users sign in using a password, restart the SSH service:

```
sudo systemctl restart ssh
```

### Configure automatic home directory creation

To enable automatic creation of the home directory when a user first signs in, complete the following steps:

- Open the `/etc/pam.d/common-session` file in an editor:

```
sudo vi /etc/pam.d/common-session
```

- Add the following line in this file below the line `session optional pam_sss.so`:

```
session required pam_mkhomedir.so skel=/etc/skel/ umask=0077
```

When done, save and exit the `common-session` file using the `:wq` command of the editor.

### Grant the 'AAD DC Administrators' group sudo privileges

To grant members of the `AAD DC Administrators` group administrative privileges on the Ubuntu VM, you add an entry to the `/etc/sudoers`. Once added, members of the `AAD DC Administrators` group can use the `sudo` command on the Ubuntu VM.

- Open the `sudoers` file for editing:

```
sudo visudo
```

- Add the following entry to the end of `/etc/sudoers` file:

```
# Add 'AAD DC Administrators' group members as admins.  
%AAD\ DC\ Administrators ALL=(ALL) NOPASSWD:ALL
```

When done, save and exit the editor using the `ctrl-X` command.

## Sign in to the VM using a domain account

To verify that the VM has been successfully joined to the managed domain, start a new SSH connection using a domain user account. Confirm that a home directory has been created, and that group membership from the domain is applied.

- Create a new SSH connection from your console. Use a domain account that belongs to the managed domain using the `ssh -l` command, such as `contosoadmin@aaddscontoso.com` and then enter the address of your VM, such as `ubuntu.aaddscontoso.com`. If you use the Azure Cloud Shell, use the public IP address of the VM rather than the internal DNS name.

```
ssh -l contosoadmin@AADDSCONTOSO.com ubuntu.aaddscontoso.com
```

- When you've successfully connected to the VM, verify that the home directory was initialized correctly:

```
pwd
```

You should be in the `/home` directory with your own directory that matches the user account.

3. Now check that the group memberships are being resolved correctly:

```
id
```

You should see your group memberships from the managed domain.

4. If you signed in to the VM as a member of the *AAD DC Administrators* group, check that you can correctly use the `sudo` command:

```
sudo apt-get update
```

## Next steps

If you have problems connecting the VM to the managed domain or signing in with a domain account, see [Troubleshooting domain join issues](#).

# Configure managed identities for Azure resources on a VM using the Azure portal

9/21/2022 • 3 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provides Azure services with an automatically managed identity in Azure Active Directory. You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, you learn how to enable and disable system and user-assigned managed identities for an Azure Virtual Machine (VM), using the Azure portal.

## Prerequisites

- If you're unfamiliar with managed identities for Azure resources, check out the [overview section](#).
- If you don't already have an Azure account, [sign up for a free account](#) before continuing.

## System-assigned managed identity

In this section, you learn how to enable and disable the system-assigned managed identity for VM using the Azure portal.

### **Enable system-assigned managed identity during creation of a VM**

To enable system-assigned managed identity on a VM during its creation, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

- Under the **Management** tab in the **Identity** section, switch **Managed service identity** to **On**.

## Create a virtual machine

Basics Disks Networking **Management** Guest config Tags Review + create

Configure monitoring and management options for your VM.

**MONITORING**

Boot diagnostics [i](#)  On  Off

OS guest diagnostics [i](#)  On  Off

\* Diagnostics storage account [i](#)  [Create new](#)

**IDENTITY**

Managed service identity [i](#)  On  Off

**AUTO-SHUTDOWN**

Enable auto-shutdown [i](#)  On  Off

**BACKUP**

Enable backup [i](#)  On  Off

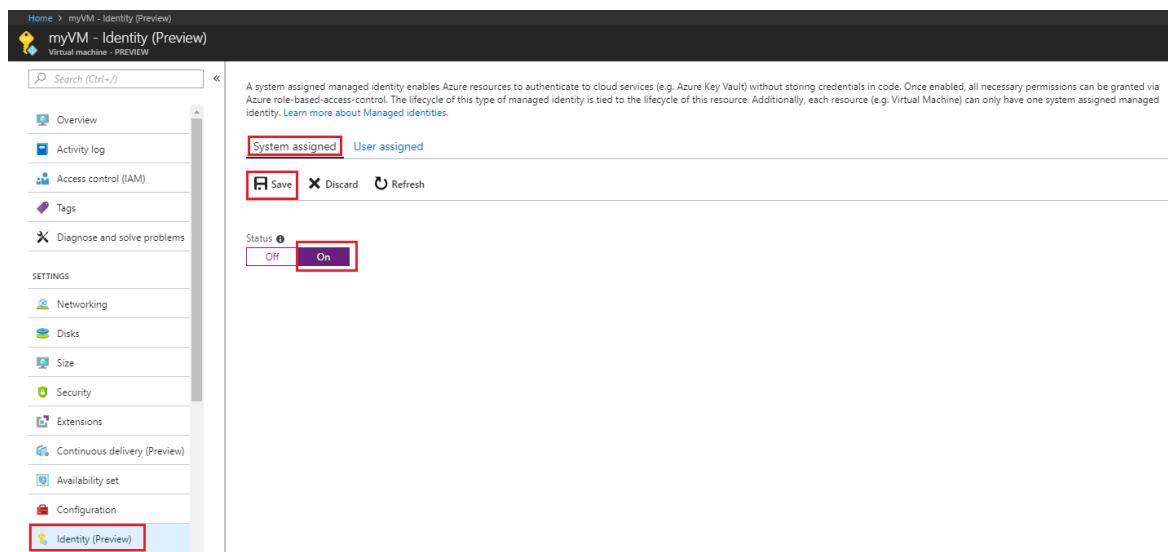
Refer to the following Quickstarts to create a VM:

- [Create a Windows virtual machine with the Azure portal](#)
- [Create a Linux virtual machine with the Azure portal](#)

### Enable system-assigned managed identity on an existing VM

To enable system-assigned managed identity on a VM that was originally provisioned without it, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Sign in to the [Azure portal](#) using an account associated with the Azure subscription that contains the VM.
2. Navigate to the desired Virtual Machine and select **Identity**.
3. Under **System assigned, Status**, select **On** and then click **Save**:



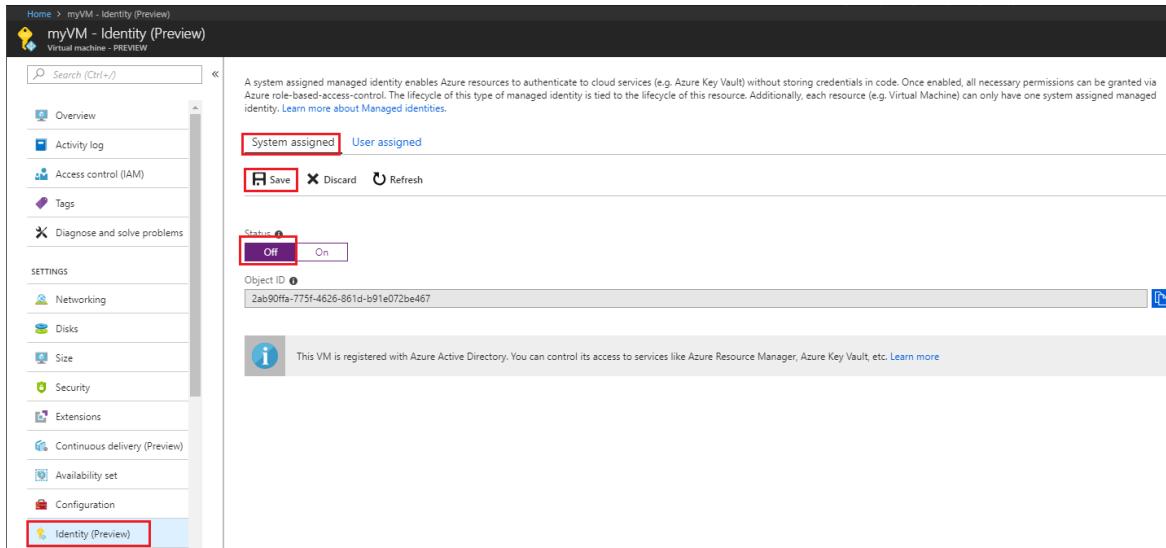
### Remove system-assigned managed identity from a VM

To remove system-assigned managed identity from a VM, your account needs the [Virtual Machine Contributor](#)

role assignment. No other Azure AD directory role assignments are required.

If you have a Virtual Machine that no longer needs system-assigned managed identity:

1. Sign in to the [Azure portal](#) using an account associated with the Azure subscription that contains the VM.
2. Navigate to the desired Virtual Machine and select **Identity**.
3. Under **System assigned, Status**, select **Off** and then click **Save**:



## User-assigned managed identity

In this section, you learn how to add and remove a user-assigned managed identity from a VM using the Azure portal.

### Assign a user-assigned identity during the creation of a VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

Currently, the Azure portal does not support assigning a user-assigned managed identity during the creation of a VM. Instead, refer to one of the following VM creation Quickstart articles to first create a VM, and then proceed to the next section for details on assigning a user-assigned managed identity to the VM:

- [Create a Windows virtual machine with the Azure portal](#)
- [Create a Linux virtual machine with the Azure portal](#)

### Assign a user-assigned managed identity to an existing VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Sign in to the [Azure portal](#) using an account associated with the Azure subscription that contains the VM.
2. Navigate to the desired VM and click **Identity, User assigned** and then **+ Add**.

User assigned managed identities enable Azure resources to authenticate to cloud services (e.g. Azure Key Vault) without storing credentials in code. This type of managed identities are created as standalone Azure resources, and have their own lifecycle. A single resource (e.g. Virtual Machine) can utilize multiple user assigned managed identities. Similarly, a single user assigned managed identity can be shared across multiple resources (e.g. Virtual Machine). [Learn more about Managed identities.](#)

**System assigned** **User assigned**

**+ Add** **Remove** **Refresh**

NAME	RESOURCE GROUP	SUBSCRIPTION
No user assigned managed identities found on this resource. Click add to get started.		

3. Click the user-assigned identity you want to add to the VM and then click **Add**.

Add user assigned managed identity  
PREVIEW

\* Subscription  
Woodgrove IT Production Environment

User assigned managed identities  
Filter by identity name and/or resource group name

ID1 Resource Group: TestRG
cyibarravmexid Resource Group: cyibarratester
devtestmsi Resource Group: DevTest
platApplIdentity Resource Group: DevTest
MKTG-UA-01 Resource Group: MARKETING-PROD
MKTG-UA-02 Resource Group: MARKETING-PROD

Selected identities:

ID1 TestRG
---------------

**Add**

## Remove a user-assigned managed identity from a VM

To remove a user-assigned identity from a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Sign in to the [Azure portal](#) using an account associated with the Azure subscription that contains the VM.

2. Navigate to the desired VM and click **Identity**, **User assigned**, the name of the user-assigned managed identity you want to delete and then click **Remove** (click **Yes** in the confirmation pane).

The screenshot shows the Azure portal interface for managing identities on a virtual machine. The left sidebar has a 'Identity' section with several options: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Networking, Disks, Size, Security, Extensions, Continuous delivery (Preview), Availability set, Configuration, and Identity. The 'Identity' option is highlighted with a red box. The main content area is titled 'myVM - Identity' and shows the 'Virtual machine - PREVIEW' section. It displays information about user-assigned managed identities, mentioning they enable authentication to cloud services without storing credentials in code. Below this, there are tabs for 'System assigned' and 'User assigned', with 'User assigned' being the active tab. A table lists managed identities with columns for NAME, RESOURCE GROUP, and SUBSCRIPTION. One entry, 'ID1', is selected and has a checkmark in the first column. The 'Remove' button in the table header is also highlighted with a red box.

NAME	RESOURCE GROUP	SUBSCRIPTION
ID1	TestRG	<SUBSCRIPTION ID>

## Next steps

- Using the Azure portal, give an Azure VM's managed identity [access to another Azure resource](#).

# Configure managed identities for Azure resources on an Azure VM using Azure CLI

9/21/2022 • 7 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provide Azure services with an automatically managed identity in Azure Active Directory. You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, using the Azure CLI, you learn how to perform the following managed identities for Azure resources operations on an Azure VM:

- Enable and disable the system-assigned managed identity on an Azure VM
- Add and remove a user-assigned managed identity on an Azure VM

If you don't already have an Azure account, [sign up for a free account](#) before continuing.

## Prerequisites

- If you're unfamiliar with managed identities for Azure resources, see [What are managed identities for Azure resources?](#). To learn about system-assigned and user-assigned managed identity types, see [Managed identity types](#).

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## System-assigned managed identity

In this section, you learn how to enable and disable the system-assigned managed identity on an Azure VM using Azure CLI.

### Enable system-assigned managed identity during creation of an Azure VM

To create an Azure VM with the system-assigned managed identity enabled, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Create a [resource group](#) for containment and deployment of your VM and its related resources, using `az group create`. You can skip this step if you already have resource group you would like to use instead:

```
az group create --name myResourceGroup --location westus
```

2. Create a VM using `az vm create`. The following example creates a VM named *myVM* with a system-

assigned managed identity, as requested by the `--assign-identity` parameter, with the specified `--role` and `--scope`. The `--admin-username` and `--admin-password` parameters specify the administrative user name and password account for virtual machine sign-in. Update these values as appropriate for your environment:

```
az vm create --resource-group myResourceGroup --name myVM --image win2016datacenter --generate-ssh-keys --assign-identity --role contributor --scope mySubscription --admin-username azureuser --admin-password myPassword12
```

### Enable system-assigned managed identity on an existing Azure VM

To enable system-assigned managed identity on a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. If you're using the Azure CLI in a local console, first sign in to Azure using `az login`. Use an account that is associated with the Azure subscription that contains the VM.

```
az login
```

2. Use `az vm identity assign` with the `identity assign` command enable the system-assigned identity to an existing VM:

```
az vm identity assign -g myResourceGroup -n myVm
```

### Disable system-assigned identity from an Azure VM

To disable system-assigned managed identity on a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

If you have a Virtual Machine that no longer needs the system-assigned identity, but still needs user-assigned identities, use the following command:

```
az vm update -n myVM -g myResourceGroup --set identity.type='UserAssigned'
```

If you have a virtual machine that no longer needs system-assigned identity and it has no user-assigned identities, use the following command:

#### NOTE

The value `none` is case sensitive. It must be lowercase.

```
az vm update -n myVM -g myResourceGroup --set identity.type="none"
```

## User-assigned managed identity

In this section, you will learn how to add and remove a user-assigned managed identity from an Azure VM using Azure CLI. If you create your user-assigned managed identity in a different RG than your VM. You'll have to use the URL of your managed identity to assign it to your VM. For example:

```
--identities  
"/subscriptions/<SUBID>/resourcegroups/<RESOURCEGROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER_ASSIGNED_IDENTITY_NAME>"
```

### Assign a user-assigned managed identity during the creation of an Azure VM

To assign a user-assigned identity to a VM during its creation, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. You can skip this step if you already have a resource group you would like to use. Create a [resource group](#) for containment and deployment of your user-assigned managed identity, using `az group create`. Be sure to replace the `<RESOURCE GROUP>` and `<LOCATION>` parameter values with your own values.:

```
az group create --name <RESOURCE GROUP> --location <LOCATION>
```

2. Create a user-assigned managed identity using [az identity create](#). The `-g` parameter specifies the resource group where the user-assigned managed identity is created, and the `-n` parameter specifies its name.

**IMPORTANT**

When you create user-assigned managed identities, only alphanumeric characters (0-9, a-z, and A-Z) and the hyphen (-) are supported. For the assignment to a virtual machine or virtual machine scale set to work properly, the name is limited to 24 characters. For more information, see [FAQs and known issues](#).

```
az identity create -g myResourceGroup -n myUserAssignedIdentity
```

The response contains details for the user-assigned managed identity created, similar to the following. The resource ID value assigned to the user-assigned managed identity is used in the following step.

```
{  
    "clientId": "73444643-8088-4d70-9532-c3a0fdc190fz",  
    "clientSecretUrl": "https://control-westcentralus.identity.azure.net/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESOURCE GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<myUserAssignedIdentity>/credential s?tid=5678&oid=9012&aid=73444643-8088-4d70-9532-c3a0fdc190fz",  
    "id": "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESOURCE GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER ASSIGNED IDENTITY NAME>",  
    "location": "westcentralus",  
    "name": "<USER ASSIGNED IDENTITY NAME>",  
    "principalId": "e5fdfdc1-ed84-4d48-8551-fe9fb9dedf11",  
    "resourceGroup": "<RESOURCE GROUP>",  
    "tags": {},  
    "tenantId": "733a8f0e-ec41-4e69-8ad8-971fc4b533b1",  
    "type": "Microsoft.ManagedIdentity/userAssignedIdentities"  
}
```

3. Create a VM using [az vm create](#). The following example creates a VM associated with the new user-assigned identity, as specified by the `--assign-identity` parameter, with the specified `--role` and `--scope`. Be sure to replace the `<RESOURCE GROUP>`, `<VM NAME>`, `<USER NAME>`, `<PASSWORD>`, `<USER ASSIGNED IDENTITY NAME>`, `<ROLE>`, and `<SUBSCRIPTION>` parameter values with your own values.

```
az vm create --resource-group <RESOURCE GROUP> --name <VM NAME> --image UbuntuLTS --admin-username <USER NAME> --admin-password <PASSWORD> --assign-identity <USER ASSIGNED IDENTITY NAME> --role <ROLE> --scope <SUBSCRIPTION>
```

### Assign a user-assigned managed identity to an existing Azure VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Create a user-assigned identity using [az identity create](#). The `-g` parameter specifies the resource group where the user-assigned identity is created, and the `-n` parameter specifies its name. Be sure to replace the `<RESOURCE GROUP>` and `<USER ASSIGNED IDENTITY NAME>` parameter values with your own values:

**IMPORTANT**

Creating user-assigned managed identities with special characters (i.e. underscore) in the name is not currently supported. Please use alphanumeric characters. Check back for updates. For more information, see [FAQs and known issues](#)

```
az identity create -g <RESOURCE GROUP> -n <USER ASSIGNED IDENTITY NAME>
```

The response contains details for the user-assigned managed identity created, similar to the following.

```
{
  "clientId": "73444643-8088-4d70-9532-c3a0fdc190fz",
  "clientSecretUrl": "https://control-westcentralus.identity.azure.net/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESOURCE GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER ASSIGNED IDENTITY NAME>/credentials?tid=5678&oid=9012&aid=73444643-8088-4d70-9532-c3a0fdc190fz",
  "id": "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESOURCE GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER ASSIGNED IDENTITY NAME>",
  "location": "westcentralus",
  "name": "<USER ASSIGNED IDENTITY NAME>",
  "principalId": "e5fdfdc1-ed84-4d48-8551-fe9fb9dedf11",
  "resourceGroup": "<RESOURCE GROUP>",
  "tags": {},
  "tenantId": "733a8f0e-ec41-4e69-8ad8-971fc4b533b1",
  "type": "Microsoft.ManagedIdentity/userAssignedIdentities"
}
```

- Assign the user-assigned identity to your VM using `az vm identity assign`. Be sure to replace the `<RESOURCE GROUP>` and `<VM NAME>` parameter values with your own values. The `<USER ASSIGNED IDENTITY NAME>` is the user-assigned managed identity's resource `name` property, as created in the previous step. If you created your user-assigned managed identity in a different RG than your VM. You'll have to use the URL of your managed identity.

```
az vm identity assign -g <RESOURCE GROUP> -n <VM NAME> --identities <USER ASSIGNED IDENTITY>
```

### Remove a user-assigned managed identity from an Azure VM

To remove a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) role assignment.

If this is the only user-assigned managed identity assigned to the virtual machine, `UserAssigned` will be removed from the identity type value. Be sure to replace the `<RESOURCE GROUP>` and `<VM NAME>` parameter values with your own values. The `<USER ASSIGNED IDENTITY>` will be the user-assigned identity's `name` property, which can be found in the identity section of the virtual machine using `az vm identity show`:

```
az vm identity remove -g <RESOURCE GROUP> -n <VM NAME> --identities <USER ASSIGNED IDENTITY>
```

If your VM does not have a system-assigned managed identity and you want to remove all user-assigned identities from it, use the following command:

#### NOTE

The value `none` is case sensitive. It must be lowercase.

```
az vm update -n myVM -g myResourceGroup --set identity.type="none" identity.userAssignedIdentities=null
```

If your VM has both system-assigned and user-assigned identities, you can remove all the user-assigned identities by switching to use only system-assigned. Use the following command:

```
az vm update -n myVM -g myResourceGroup --set identity.type='SystemAssigned'
identity.userAssignedIdentities=null
```

## Next steps

- [Managed identities for Azure resources overview](#)
- For the full Azure VM creation Quickstarts, see:
  - [Create a Windows virtual machine with CLI](#)
  - [Create a Linux virtual machine with CLI](#)

# Configure managed identities for Azure resources on an Azure VM using PowerShell

9/21/2022 • 6 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provide Azure services with an automatically managed identity in Azure Active Directory. You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, using PowerShell, you learn how to perform the following managed identities for Azure resources operations on an Azure VM.

## NOTE

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az](#).

## Prerequisites

- If you're unfamiliar with managed identities for Azure resources, check out the [overview section](#). Be sure to review the [difference between a system-assigned and user-assigned managed identity](#).
- If you don't already have an Azure account, [sign up for a free account](#) before continuing.
- To run the example scripts, you have two options:
  - Use the [Azure Cloud Shell](#), which you can open using the Try It button on the top-right corner of code blocks.
  - Run scripts locally by installing the latest version of [Azure PowerShell](#), then sign in to Azure using `Connect-AzAccount`.

## System-assigned managed identity

In this section, you'll learn how to enable and disable the system-assigned managed identity using Azure PowerShell.

### Enable system-assigned managed identity during creation of an Azure VM

To create an Azure VM with the system-assigned managed identity enabled, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Refer to one of the following Azure VM Quickstarts, completing only the necessary sections ("Sign in to Azure", "Create resource group", "Create networking group", "Create the VM").

When you get to the "Create the VM" section, make a slight modification to the `New-AzVMConfig` cmdlet syntax. Be sure to add a `-IdentityType SystemAssigned` parameter to provision the VM with the system-assigned identity enabled, for example:

```
$vmConfig = New-AzVMConfig -VMName myVM -IdentityType SystemAssigned ...
```

- Create a Windows virtual machine using PowerShell
- Create a Linux virtual machine using PowerShell

## Enable system-assigned managed identity on an existing Azure VM

To enable system-assigned managed identity on a VM that was originally provisioned without it, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Retrieve the VM properties using the `Get-AzVM` cmdlet. Then to enable a system-assigned managed identity, use the `-IdentityType` switch on the `Update-AzVM` cmdlet:

```
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM
Update-AzVM -ResourceGroupName myResourceGroup -VM $vm -IdentityType SystemAssigned
```

## Add VM system assigned identity to a group

After you have enabled system assigned identity on a VM, you can add it to a group. The following procedure adds a VM's system assigned identity to a group.

1. Retrieve and note the `ObjectId` (as specified in the `Id` field of the returned values) of the VM's service principal:

```
Get-AzADServicePrincipal -displayname "myVM"
```

2. Retrieve and note the `ObjectId` (as specified in the `Id` field of the returned values) of the group:

```
Get-AzADGroup -searchstring "myGroup"
```

3. Add the VM's service principal to the group:

```
Add-AzureADGroupMember -ObjectId "<objectId of group>" -RefObjectId "<object id of VM service principal>"
```

## Disable system-assigned managed identity from an Azure VM

To disable system-assigned managed identity on a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

If you have a Virtual Machine that no longer needs the system-assigned managed identity but still needs user-assigned managed identities, use the following cmdlet:

1. Retrieve the VM properties using the `Get-AzVM` cmdlet and set the `-IdentityType` parameter to `UserAssigned`:

```
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM
Update-AzVm -ResourceGroupName myResourceGroup -VM $vm -IdentityType "UserAssigned"
```

If you have a virtual machine that no longer needs system-assigned managed identity and it has no user-assigned managed identities, use the following commands:

```
$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM
Update-AzVm -ResourceGroupName myResourceGroup -VM $vm -IdentityType None
```

# User-assigned managed identity

In this section, you learn how to add and remove a user-assigned managed identity from a VM using Azure PowerShell.

## Assign a user-assigned managed identity to a VM during creation

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Refer to one of the following Azure VM Quickstarts, completing only the necessary sections ("Sign in to Azure", "Create resource group", "Create networking group", "Create the VM").

When you get to the "Create the VM" section, make a slight modification to the `New-AzVMConfig` cmdlet syntax. Add the `-IdentityType UserAssigned` and `-IdentityID` parameters to provision the VM with a user-assigned identity. Replace `<VM NAME>`, `<SUBSCRIPTION ID>`, `<RESOURCE GROUP>`, and `<USER ASSIGNED IDENTITY NAME>` with your own values. For example:

```
$vmConfig = New-AzVMConfig -VMName <VM NAME> -IdentityType UserAssigned -IdentityID  
"/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESOURCE  
GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER ASSIGNED IDENTITY NAME>..."
```

- [Create a Windows virtual machine using PowerShell](#)
- [Create a Linux virtual machine using PowerShell](#)

## Assign a user-assigned managed identity to an existing Azure VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Create a user-assigned managed identity using the `New-AzUserAssignedIdentity` cmdlet. Note the `Id` in the output because you'll need this information in the next step.

### IMPORTANT

Creating user-assigned managed identities only supports alphanumeric, underscore and hyphen (0-9 or a-z or A-Z, \_ or -) characters. Additionally, name should be limited from 3 to 128 character length for the assignment to VM/VMSS to work properly. For more information, see [FAQs and known issues](#)

```
New-AzUserAssignedIdentity -ResourceGroupName <RESOURCEGROUP> -Name <USER ASSIGNED IDENTITY NAME>
```

2. Retrieve the VM properties using the `Get-AzVM` cmdlet. Then to assign a user-assigned managed identity to the Azure VM, use the `-IdentityType` and `-IdentityID` switch on the `Update-AzVM` cmdlet. The value for the `-IdentityId` parameter is the `Id` you noted in the previous step. Replace `<VM NAME>`, `<SUBSCRIPTION ID>`, `<RESOURCE GROUP>`, and `<USER ASSIGNED IDENTITY NAME>` with your own values.

### WARNING

To retain any previously user-assigned managed identities assigned to the VM, query the `Identity` property of the VM object (for example, `$vm.Identity`). If any user assigned managed identities are returned, include them in the following command along with the new user assigned managed identity you would like to assign to the VM.

```
$vm = Get-AzVM -ResourceGroupName <RESOURCE GROUP> -Name <VM NAME>
Update-AzVM -ResourceGroupName <RESOURCE GROUP> -VM $vm -IdentityType UserAssigned -IdentityID
"/subscriptions/<SUBSCRIPTION ID>/resourcegroups/<RESROUCE
GROUP>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<USER ASSIGNED IDENTITY NAME>"
```

## Remove a user-assigned managed identity from an Azure VM

To remove a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) role assignment.

If your VM has multiple user-assigned managed identities, you can remove all but the last one using the following commands. Be sure to replace the `<RESOURCE GROUP>` and `<VM NAME>` parameter values with your own values. The `<USER ASSIGNED IDENTITY NAME>` is the user-assigned managed identity's name property, which should remain on the VM. This information can be found by querying the `Identity` property of the VM object. For example, `$vm.Identity`:

```
$vm = Get-AzVm -ResourceGroupName myResourceGroup -Name myVm
Update-AzVm -ResourceGroupName myResourceGroup -VirtualMachine $vm -IdentityType UserAssigned -IdentityID
<USER ASSIGNED IDENTITY NAME>
```

If your VM doesn't have a system-assigned managed identity and you want to remove all user-assigned managed identities from it, use the following command:

```
$vm = Get-AzVm -ResourceGroupName myResourceGroup -Name myVm
Update-AzVm -ResourceGroupName myResourceGroup -VM $vm -IdentityType None
```

If your VM has both system-assigned and user-assigned managed identities, you can remove all the user-assigned managed identities by switching to use only system-assigned managed identities.

```
$vm = Get-AzVm -ResourceGroupName myResourceGroup -Name myVm
Update-AzVm -ResourceGroupName myResourceGroup -VirtualMachine $vm -IdentityType "SystemAssigned"
```

## Next steps

- [Managed identities for Azure resources overview](#)
- For the full Azure VM creation Quickstarts, see:
  - [Create a Windows virtual machine with PowerShell](#)
  - [Create a Linux virtual machine with PowerShell](#)

# Configure managed identities for Azure resources on an Azure VM using templates

9/21/2022 • 6 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provide Azure services with an automatically managed identity in Azure Active Directory. You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, using the Azure Resource Manager deployment template, you learn how to perform the following managed identities for Azure resources operations on an Azure VM:

## Prerequisites

- If you're unfamiliar with using Azure Resource Manager deployment template, check out the [overview section](#). Be sure to review the [difference between a system-assigned and user-assigned managed identity](#).
- If you don't already have an Azure account, [sign up for a free account](#) before continuing.

## Azure Resource Manager templates

As with the Azure portal and scripting, [Azure Resource Manager](#) templates allow you to deploy new or modified resources defined by an Azure resource group. Several options are available for template editing and deployment, both local and portal-based, including:

- Using a [custom template from the Azure Marketplace](#), which allows you to create a template from scratch, or base it on an existing common or [quickstart template](#).
- Deriving from an existing resource group, by exporting a template from either [the original deployment](#), or from the [current state of the deployment](#).
- Using a local [JSON editor](#) (such as VS Code), and then uploading and deploying by using PowerShell or CLI.
- Using the Visual Studio [Azure Resource Group project](#) to both create and deploy a template.

Regardless of the option you choose, template syntax is the same during initial deployment and redeployment. Enabling a system or user-assigned managed identity on a new or existing VM is done in the same manner. Also, by default, Azure Resource Manager does an [incremental update](#) to deployments.

## System-assigned managed identity

In this section, you will enable and disable a system-assigned managed identity using an Azure Resource Manager template.

### **Enable system-assigned managed identity during creation of an Azure VM or on an existing VM**

To enable system-assigned managed identity on a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Whether you sign in to Azure locally or via the Azure portal, use an account that is associated with the Azure subscription that contains the VM.

2. To enable system-assigned managed identity, load the template into an editor, locate the `Microsoft.Compute/virtualMachines` resource of interest within the `resources` section and add the `"identity"` property at the same level as the `"type": "Microsoft.Compute/virtualMachines"` property. Use the following syntax:

```
"identity": {  
    "type": "SystemAssigned"  
},
```

3. When you're done, the following sections should be added to the `resource` section of your template and it should resemble the following:

```
"resources": [  
    {  
        //other resource provider properties...  
        "apiVersion": "2018-06-01",  
        "type": "Microsoft.Compute/virtualMachines",  
        "name": "[variables('vmName')]",  
        "location": "[resourceGroup().location]",  
        "identity": {  
            "type": "SystemAssigned",  
        }  
    }  
]
```

## Assign a role the VM's system-assigned managed identity

After you enable a system-assigned managed identity on your VM, you may want to grant it a role such as **Reader** access to the resource group in which it was created. You can find detailed information to help you with this step in the [Assign Azure roles using Azure Resource Manager templates](#) article.

## Disable a system-assigned managed identity from an Azure VM

To remove system-assigned managed identity from a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Whether you sign in to Azure locally or via the Azure portal, use an account that is associated with the Azure subscription that contains the VM.
2. Load the template into an [editor](#) and locate the `Microsoft.Compute/virtualMachines` resource of interest within the `resources` section. If you have a VM that only has system-assigned managed identity, you can disable it by changing the identity type to `None`.

### Microsoft.Compute/virtualMachines API version 2018-06-01

If your VM has both system and user-assigned managed identities, remove `SystemAssigned` from the identity type and keep `UserAssigned` along with the `userAssignedIdentities` dictionary values.

### Microsoft.Compute/virtualMachines API version 2018-06-01

If your `apiVersion` is `2017-12-01` and your VM has both system and user-assigned managed identities, remove `SystemAssigned` from the identity type and keep `UserAssigned` along with the `identityIds` array of the user-assigned managed identities.

The following example shows you how to remove a system-assigned managed identity from a VM with no user-assigned managed identities:

```
{  
    "apiVersion": "2018-06-01",  
    "type": "Microsoft.Compute/virtualMachines",  
    "name": "[parameters('vmName')]",  
    "location": "[resourceGroup().location]",  
    "identity": {  
        "type": "None"  
    }  
}
```

## User-assigned managed identity

In this section, you assign a user-assigned managed identity to an Azure VM using Azure Resource Manager template.

### NOTE

To create a user-assigned managed identity using an Azure Resource Manager Template, see [Create a user-assigned managed identity](#).

### Assign a user-assigned managed identity to an Azure VM

To assign a user-assigned identity to a VM, your account needs the [Managed Identity Operator](#) role assignment. No other Azure AD directory role assignments are required.

- Under the `resources` element, add the following entry to assign a user-assigned managed identity to your VM. Be sure to replace `<USERASSIGNEDIDENTITY>` with the name of the user-assigned managed identity you created.

#### Microsoft.Compute/virtualMachines API version 2018-06-01

If your `apiVersion` is `2018-06-01`, your user-assigned managed identities are stored in the `userAssignedIdentities` dictionary format and the `<USERASSIGNEDIDENTITYNAME>` value must be stored in a variable defined in the `variables` section of your template.

```
{  
    "apiVersion": "2018-06-01",  
    "type": "Microsoft.Compute/virtualMachines",  
    "name": "[variables('vmName')]",  
    "location": "[resourceGroup().location]",  
    "identity": {  
        "type": "userAssigned",  
        "userAssignedIdentities": {  
            "  
[resourceID('Microsoft.ManagedIdentity/userAssignedIdentities/',variables('<USERASSIGNEDIDENTITYNAME>  
''))]": {}  
        }  
    }  
}
```

#### Microsoft.Compute/virtualMachines API version 2017-12-01

If your `apiVersion` is `2017-12-01`, your user-assigned managed identities are stored in the `identityIds` array and the `<USERASSIGNEDIDENTITYNAME>` value must be stored in a variable defined in the `variables` section of your template.

```
{
    "apiVersion": "2017-12-01",
    "type": "Microsoft.Compute/virtualMachines",
    "name": "[variables('vmName')]",
    "location": "[resourceGroup().location]",
    "identity": {
        "type": "userAssigned",
        "identityIds": [
            ""
        ]
    }
}
```

2. When you're done, the following sections should be added to the `resource` section of your template and it should resemble the following:

#### **Microsoft.Compute/virtualMachines API version 2018-06-01**

```
"resources": [
{
    //other resource provider properties...
    "apiVersion": "2018-06-01",
    "type": "Microsoft.Compute/virtualMachines",
    "name": "[variables('vmName')]",
    "location": "[resourceGroup().location]",
    "identity": {
        "type": "userAssigned",
        "userAssignedIdentities": {
            ""
        }
    }
}
]
```

#### **Microsoft.Compute/virtualMachines API version 2017-12-01**

```
"resources": [
{
    //other resource provider properties...
    "apiVersion": "2017-12-01",
    "type": "Microsoft.Compute/virtualMachines",
    "name": "[variables('vmName')]",
    "location": "[resourceGroup().location]",
    "identity": {
        "type": "userAssigned",
        "identityIds": [
            ""
        ]
    }
}
]
```

### **Remove a user-assigned managed identity from an Azure VM**

To remove a user-assigned identity from a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Whether you sign in to Azure locally or via the Azure portal, use an account that is associated with the Azure subscription that contains the VM.
2. Load the template into an [editor](#) and locate the `Microsoft.Compute/virtualMachines` resource of interest within the `resources` section. If you have a VM that only has user-assigned managed identity, you can disable it by changing the identity type to `None`.

The following example shows you how to remove all user-assigned managed identities from a VM with no system-assigned managed identities:

```
{  
    "apiVersion": "2018-06-01",  
    "type": "Microsoft.Compute/virtualMachines",  
    "name": "[parameters('vmName')]",  
    "location": "[resourceGroup().location]",  
    "identity": {  
        "type": "None"  
    },  
}
```

#### **Microsoft.Compute/virtualMachines API version 2018-06-01**

To remove a single user-assigned managed identity from a VM, remove it from the `userAssignedIdentities` dictionary.

If you have a system-assigned managed identity, keep it in the `type` value under the `identity` value.

#### **Microsoft.Compute/virtualMachines API version 2017-12-01**

To remove a single user-assigned managed identity from a VM, remove it from the `identityIds` array.

If you have a system-assigned managed identity, keep it in the `type` value under the `identity` value.

## Next steps

- [Managed identities for Azure resources overview](#).

# Configure Managed identities for Azure resources on an Azure VM using REST API calls

9/21/2022 • 14 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provide Azure services with an automatically managed system identity in Azure Active Directory. You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, using CURL to make calls to the Azure Resource Manager REST endpoint, you learn how to perform the following managed identities for Azure resources operations on an Azure VM:

- Enable and disable the system-assigned managed identity on an Azure VM
- Add and remove a user-assigned managed identity on an Azure VM

If you don't already have an Azure account, [sign up for a free account](#) before continuing.

## Prerequisites

- If you're unfamiliar with managed identities for Azure resources, see [What are managed identities for Azure resources?](#). To learn about system-assigned and user-assigned managed identity types, see [Managed identity types](#).

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## System-assigned managed identity

In this section, you learn how to enable and disable system-assigned managed identity on an Azure VM using CURL to make calls to the Azure Resource Manager REST endpoint.

### **Enable system-assigned managed identity during creation of an Azure VM**

To create an Azure VM with the system-assigned managed identity enabled, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Create a [resource group](#) for containment and deployment of your VM and its related resources, using [az group create](#). You can skip this step if you already have resource group you would like to use instead:

```
az group create --name myResourceGroup --location westus
```

2. Create a [network interface](#) for your VM:

```
az network nic create -g myResourceGroup --vnet-name myVnet --subnet mySubnet -n myNic
```

3. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

4. Using Azure Cloud Shell, create a VM using CURL to call the Azure Resource Manager REST endpoint. The following example creates a VM named *myVM* with a system-assigned managed identity, as identified in the request body by the value `"identity": {"type": "SystemAssigned"}`. Replace `<ACCESS TOKEN>` with the value you received in the previous step when you requested a Bearer access token and the `<SUBSCRIPTION ID>` value as appropriate for your environment.

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01' -X PUT -d '{"location": "westus", "name": "myVM", "identity": {"type": "SystemAssigned"}, "properties": {"hardwareProfile": {"vmSize": "Standard_D2_v2"}, "storageProfile": {"imageReference": {"sku": "2016-Datacenter", "publisher": "MicrosoftWindowsServer", "version": "latest", "offer": "WindowsServer"}, "osDisk": {"caching": "ReadWrite", "managedDisk": {"storageAccountType": "StandardSSD_LRS"}, "name": "myVM3osdisk", "createOption": "FromImage"}, "dataDisks": [{"diskSizeGB": 1023, "createOption": "Empty", "lun": 0}, {"diskSizeGB": 1023, "createOption": "Empty", "lun": 1}], "osProfile": {"adminUsername": "azureuser", "computerName": "myVM", "adminPassword": "<SECURE PASSWORD STRING>"}, "networkProfile": {"networkInterfaces": [{"id": "/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic", "properties": {"primary": true}}]} }' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PUT https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "location": "westus",
  "name": "myVM",
  "identity": {
    "type": "SystemAssigned"
  },
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_D2_v2"
    },
    "storageProfile": {
      "imageReference": {
        "sku": "2016-Datacenter",
        "publisher": "MicrosoftWindowsServer",
        "version": "latest",
        "offer": "WindowsServer"
      },
      "osDisk": {
        "caching": "ReadWrite",
        "managedDisk": {
          "storageAccountType": "StandardSSD_LRS"
        },
        "name": "myVM3osdisk",
        "createOption": "FromImage"
      },
      "dataDisks": [
        {
          "lun": 0,
          "createOption": "Empty",
          "diskSizeGB": 1023
        },
        {
          "lun": 1,
          "createOption": "Empty",
          "diskSizeGB": 1023
        }
      ]
    },
    "osProfile": {
      "adminUsername": "azureuser",
      "computerName": "myVM",
      "adminPassword": "myPassword12"
    },
    "networkProfile": {
      "networkInterfaces": [
        {
          "id": "/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic",
          "properties": {
            "primary": true
          }
        }
      ]
    }
  }
}
```

## Enable system-assigned identity on an existing Azure VM

To enable system-assigned managed identity on a VM that was originally provisioned without it, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

2. Use the following CURL command to call the Azure Resource Manager REST endpoint to enable system-assigned managed identity on your VM as identified in the request body by the value

`{"identity": {"type": "SystemAssigned"}}` for a VM named *myVM*. Replace `<ACCESS TOKEN>` with the value you received in the previous step when you requested a Bearer access token and the `<SUBSCRIPTION ID>` value as appropriate for your environment.

#### IMPORTANT

To ensure you don't delete any existing user-assigned managed identities that are assigned to the VM, you need to list the user-assigned managed identities by using this CURL command:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01' -H "Authorization: Bearer <ACCESS TOKEN>"
```

If you have any user-assigned managed identities assigned to the VM as identified in the `identity` value in the response, skip to step 3 that shows you how to retain user-assigned managed identities while enabling system-assigned managed identity on your VM.

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01' -X PATCH -d '{"identity": {"type": "SystemAssigned" }}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01 HTTP/1.1
```

#### Request headers

REQUEST HEADER	DESCRIPTION
<code>Content-Type</code>	Required. Set to <code>application/json</code> .
<code>Authorization</code>	Required. Set to a valid <code>Bearer</code> access token.

#### Request body

```
{  
  "identity": {  
    "type": "SystemAssigned"  
  }  
}
```

3. To enable system-assigned managed identity on a VM with existing user-assigned managed identities, you need to add `SystemAssigned` to the `type` value.

For example, if your VM has the user-assigned managed identities `ID1` and `ID2` assigned to it, and you would like to add system-assigned managed identity to the VM, use the following CURL call. Replace `<ACCESS TOKEN>` and `<SUBSCRIPTION ID>` with values appropriate to your environment.

API version `2018-06-01` stores user-assigned managed identities in the `userAssignedIdentities` value in a dictionary format as opposed to the `identityIds` value in an array format used in API version

2017-12-01 .

## API VERSION 2018-06-01

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01' -X PATCH -d '{"identity": {"type": "SystemAssigned, UserAssigned", "userAssignedIdentities": [
{}], "/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1": [
{}], "/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2": [
{}]} }' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01 HTTP/1.1
```

### Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

### Request body

```
{
  "identity": {
    "type": "SystemAssigned, UserAssigned",
    "userAssignedIdentities": [
      "/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1": {

      },
      "/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2": {

      }
    ]
  }
}
```

## API VERSION 2017-12-01

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01' -X PATCH -d '{"identity": {"type": "SystemAssigned, UserAssigned", "identityIds": [
"/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1", "/subscriptions/<>SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2"]}}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{  
    "identity":{  
        "type":"SystemAssigned, UserAssigned",  
        "identityIds": [  
            "/subscriptions/<>SUBSCRIPTION  
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1",  
            "/subscriptions/<>SUBSCRIPTION  
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2"  
        ]  
    }  
}
```

## Disable system-assigned managed identity from an Azure VM

To disable system-assigned managed identity on a VM, your account needs the [Virtual Machine Contributor](#) role assignment. No other Azure AD directory role assignments are required.

1. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

2. Update the VM using CURL to call the Azure Resource Manager REST endpoint to disable system-assigned managed identity. The following example disables system-assigned managed identity as identified in the request body by the value `{"identity":{"type":"None"}}` from a VM named *myVM*. Replace `<ACCESS TOKEN>` with the value you received in the previous step when you requested a Bearer access token and the `<SUBSCRIPTION ID>` value as appropriate for your environment.

### IMPORTANT

To ensure you don't delete any existing user-assigned managed identities that are assigned to the VM, you need to list the user-assigned managed identities by using this CURL command:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE  
GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01' -H  
"Authorization: Bearer <ACCESS TOKEN>"
```

- If you have any user-assigned managed identities assigned to the VM as identified in the `identity` value in the response, skip to step 3 that shows you how to retain user-assigned managed identities while disabling system-assigned managed identity on your VM.

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-  
06-01' -X PATCH -d '{"identity":{"type":"None"}}' -H "Content-Type: application/json" -H  
"Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{  
    "identity":{  
        "type":"None"  
    }  
}
```

To remove system-assigned managed identity from a virtual machine that has user-assigned managed identities, remove `SystemAssigned` from the `{"identity": {"type: " "}}` value while keeping the `UserAssigned` value and the `userAssignedIdentities` dictionary values if you are using API version `2018-06-01`. If you are using API version `2017-12-01` or earlier, keep the `identityIds` array.

## User-assigned managed identity

In this section, you learn how to add and remove user-assigned managed identity on an Azure VM using CURL to make calls to the Azure Resource Manager REST endpoint.

### Assign a user-assigned managed identity during the creation of an Azure VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

2. Create a [network interface](#) for your VM:

```
az network nic create -g myResourceGroup --vnet-name myVnet --subnet mySubnet -n myNic
```

3. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

4. Create a user-assigned managed identity using the instructions found here: [Create a user-assigned managed identity](#).

5. Create a VM using CURL to call the Azure Resource Manager REST endpoint. The following example

creates a VM named *myVM* in the resource group *myResourceGroup* with a user-assigned managed identity `ID1`, as identified in the request body by the value `"identity": {"type": "UserAssigned"}`. Replace `<ACCESS TOKEN>` with the value you received in the previous step when you requested a Bearer access token and the `<SUBSCRIPTION ID>` value as appropriate for your environment.

## API VERSION 2018-06-01

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01' -X PUT -d '{"location": "westus", "name": "myVM", "identity": {"type": "UserAssigned", "identityIds": [
"/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"]}, "properties": {"hardwareProfile": {"vmSize": "Standard_D2_v2"}, "storageProfile": {"imageReference": {"sku": "2016-
Datacenter", "publisher": "MicrosoftWindowsServer", "version": "latest", "offer": "WindowsServer"}, "osDisk": {"caching": "ReadWrite", "managedDisk": {"storageAccountType": "StandardSSD_LRS"}, "name": "myVM3osdisk", "createOption": "FromImage"}, "dataDisks": [{"diskSizeGB": 1023, "createOption": "Empty", "lun": 0}, {"diskSizeGB": 1023, "createOption": "Empty", "lun": 1}], "osProfile": {"adminUsername": "azureuser", "computerName": "myVM", "adminPassword": "myPassword12"}, "networkProfile": {"networkInterfaces": [{"id": "/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic", "properties": {"primary": true}}]}]]}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PUT https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01 HTTP/1.1
```

### Request headers

REQUEST HEADER	DESCRIPTION
<code>Content-Type</code>	Required. Set to <code>application/json</code> .
<code>Authorization</code>	Required. Set to a valid <code>Bearer</code> access token.

### Request body

```
{
  "location": "westus",
  "name": "myVM",
  "identity": {
    "type": "UserAssigned",
    "identityIds": [
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"
    ]
  },
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_D2_v2"
    },
    "storageProfile": {
      "imageReference": {
        "sku": "2016-Datacenter",
        "publisher": "MicrosoftWindowsServer",
        "version": "latest",
        "offer": "WindowsServer"
      },
      "osDisk": {
        "caching": "ReadWrite",
        "managedDisk": {
          "storageAccountType": "StandardSSD_LRS"
        },
        "name": "myVM3osdisk",
        "createOption": "FromImage"
      },
      "dataDisks": [
        {
          "diskSizeGB": 1023,
          "createOption": "Empty",
          "lun": 0
        },
        {
          "diskSizeGB": 1023,
          "createOption": "Empty",
          "lun": 1
        }
      ]
    },
    "osProfile": {
      "adminUsername": "azureuser",
      "computerName": "myVM",
      "adminPassword": "myPassword12"
    },
    "networkProfile": {
      "networkInterfaces": [
        {
          "id": "/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic",
          "properties": {
            "primary": true
          }
        }
      ]
    }
  }
}
```

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-  
12-01' -X PUT -d '{"location": "westus", "name": "myVM", "identity": {"type": "UserAssigned", "identityIds":  
["/subscriptions/<SUBSCRIPTION  
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"]},  
"properties": {"hardwareProfile": {"vmSize": "Standard_D2_v2"}, "storageProfile": {"imageReference":  
{"sku": "2016-  
Datacenter", "publisher": "MicrosoftWindowsServer", "version": "latest", "offer": "WindowsServer"}, "osDisk"  
:{ "caching": "ReadWrite", "managedDisk":  
{"storageAccountType": "StandardSSD_LRS"}, "name": "myVM3osdisk", "createOption": "FromImage"}, "dataDisks"  
:[ {"diskSizeGB": 1023, "createOption": "Empty", "lun": 0},  
{"diskSizeGB": 1023, "createOption": "Empty", "lun": 1}], "osProfile":  
{"adminUsername": "azureuser", "computerName": "myVM", "adminPassword": "myPassword12"}, "networkProfile":  
{"networkInterfaces": [{"id": "/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic"}, "properties":  
{"primary": true}]}]}]' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PUT https://management.azure.com/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-  
12-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "location": "westus",
  "name": "myVM",
  "identity": {
    "type": "UserAssigned",
    "identityIds": [
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"
    ]
  },
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_D2_v2"
    },
    "storageProfile": {
      "imageReference": {
        "sku": "2016-Datacenter",
        "publisher": "MicrosoftWindowsServer",
        "version": "latest",
        "offer": "WindowsServer"
      },
      "osDisk": {
        "caching": "ReadWrite",
        "managedDisk": {
          "storageAccountType": "StandardSSD_LRS"
        },
        "name": "myVM3osdisk",
        "createOption": "FromImage"
      },
      "dataDisks": [
        {
          "diskSizeGB": 1023,
          "createOption": "Empty",
          "lun": 0
        },
        {
          "diskSizeGB": 1023,
          "createOption": "Empty",
          "lun": 1
        }
      ]
    },
    "osProfile": {
      "adminUsername": "azureuser",
      "computerName": "myVM",
      "adminPassword": "myPassword12"
    },
    "networkProfile": {
      "networkInterfaces": [
        {
          "id": "/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/myNic",
          "properties": {
            "primary": true
          }
        }
      ]
    }
  }
}
```

## Assign a user-assigned managed identity to an existing Azure VM

To assign a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) and [Managed Identity Operator](#) role assignments. No other Azure AD directory role assignments are required.

1. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

2. Create a user-assigned managed identity using the instructions found here, [Create a user-assigned managed identity](#).
3. To ensure you don't delete existing user or system-assigned managed identities that are assigned to the VM, you need to list the identity types assigned to the VM by using the following CURL command. If you have managed identities assigned to the virtual machine scale set, they are listed under in the `identity` value.

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01' -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
GET https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<code>Authorization</code>	Required. Set to a valid <code>Bearer</code> access token.

If you have any user or system-assigned managed identities assigned to the VM as identified in the `identity` value in the response, skip to step 5 that shows you how to retain the system-assigned managed identity while adding a user-assigned managed identity on your VM.

4. If you don't have any user-assigned managed identities assigned to your VM, use the following CURL command to call the Azure Resource Manager REST endpoint to assign the first user-assigned managed identity to the VM.

The following example assigns a user-assigned managed identity, `ID1` to a VM named `myVM` in the resource group `myResourceGroup`. Replace `<ACCESS TOKEN>` with the value you received in the previous step when you requested a Bearer access token and the `<SUBSCRIPTION ID>` value as appropriate for your environment.

## API VERSION 2018-06-01

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01' -X PATCH -d '{"identity":{"type":"UserAssigned", "userAssignedIdentities": [{""/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1": {}}}}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity":{
    "type":"UserAssigned",
    "userAssignedIdentities":{
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1":{

        }
      }
    }
}
```

API VERSION 2017-12-01

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01' -X PATCH -d '{"identity":{"type":"userAssigned", "identityIds":["/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"]}}'
  -H "Content-Type: application/json" -H "Authorization:Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity":{
    "type":"userAssigned",
    "identityIds":[
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"
    ]
  }
}
```

5. If you have an existing user-assigned or system-assigned managed identity assigned to your VM:

API VERSION 2018-06-01

Add the user-assigned managed identity to the `userAssignedIdentities` dictionary value.

For example, if you have system-assigned managed identity and the user-assigned managed identity `ID1` currently assigned to your VM and would like to add the user-assigned managed identity `ID2` to it:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01' -X PATCH -d '{"identity":{"type":"SystemAssigned, UserAssigned", "userAssignedIdentities":"
"/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1":{},"/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2":{}}}}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-
06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<code>Content-Type</code>	Required. Set to <code>application/json</code> .
<code>Authorization</code>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity": {
    "type": "SystemAssigned, UserAssigned",
    "userAssignedIdentities": {
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1": {

      },
      "/subscriptions/<SUBSCRIPTION
ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2": {

      }
    }
  }
}
```

## API VERSION 2017-12-01

Retain the user-assigned managed identities you would like to keep in the `identityIds` array value while adding the new user-assigned managed identity.

For example, if you have system-assigned managed identity and the user-assigned managed identity `ID1` currently assigned to your VM and would like to add the user-assigned managed identity `ID2` to it:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-12-01' -X PATCH -d '{"identity":{"type":"SystemAssigned,UserAssigned", "identityIds": ["/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1", "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2"]}}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-12-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity":{
    "type":"SystemAssigned,UserAssigned",
    "identityIds":[
      "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1",
      "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2"
    ]
  }
}
```

## Remove a user-assigned managed identity from an Azure VM

To remove a user-assigned identity to a VM, your account needs the [Virtual Machine Contributor](#) role assignment.

1. Retrieve a Bearer access token, which you will use in the next step in the Authorization header to create your VM with a system-assigned managed identity.

```
az account get-access-token
```

2. To ensure you don't delete any existing user-assigned managed identities that you would like to keep assigned to the VM or remove the system-assigned managed identity, you need to list the managed identities by using the following CURL command:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01' -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
GET https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/<RESOURCE GROUP>/providers/Microsoft.Compute/virtualMachines/<VM NAME>?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

If you have managed identities assigned to the VM, they are listed in the response in the `identity` value.

For example, if you have user-assigned managed identities `ID1` and `ID2` assigned to your VM, and you only want to keep `ID1` assigned and retain the system-assigned identity:

## API VERSION 2018-06-01

Add `null` to the user-assigned managed identity you would like to remove:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01' -X PATCH -d '{"identity": {"type": "SystemAssigned, UserAssigned", "userAssignedIdentities": {"/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2": null}}}' -H "Content-Type: application/json" -H "Authorization: Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity": {
    "type": "SystemAssigned, UserAssigned",
    "userAssignedIdentities": {
      "/subscriptions/<SUBSCRIPTION ID>/resourcegroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID2": null
    }
  }
}
```

## API VERSION 2017-12-01

Retain only the user-assigned managed identity(s) you would like to keep in the `identityIds` array:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01' -X PATCH -d '{"identity":{"type":"SystemAssigned, UserAssigned", "identityIds":
["/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"]}}'
-H "Content-Type: application/json" -H "Authorization:Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2017-
12-01 HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{
  "identity":{
    "type":"SystemAssigned, UserAssigned",
    "identityIds":[
      "/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.ManagedIdentity/userAssignedIdentities/ID1"
    ]
  }
}
```

If your VM has both system-assigned and user-assigned managed identities, you can remove all the user-assigned managed identities by switching to use only system-assigned managed identity using the following command:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01'
-X PATCH -d '{"identity":{"type":"SystemAssigned"}}' -H "Content-Type: application/json" -H
"Authorization:Bearer <ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01
HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{  
  "identity":{  
    "type":"SystemAssigned"  
  }  
}
```

If your VM has only user-assigned managed identities and you would like to remove them all, use the following command:

```
curl 'https://management.azure.com/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01'  
-X PATCH -d '{"identity":{"type":"None"}}' -H "Content-Type: application/json" -H Authorization:"Bearer  
<ACCESS TOKEN>"
```

```
PATCH https://management.azure.com/subscriptions/<SUBSCRIPTION  
ID>/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM?api-version=2018-06-01  
HTTP/1.1
```

## Request headers

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i>	Required. Set to <code>application/json</code> .
<i>Authorization</i>	Required. Set to a valid <code>Bearer</code> access token.

## Request body

```
{  
  "identity":{  
    "type":"None"  
  }  
}
```

## Next steps

For information on how to create, list, or delete user-assigned managed identities using REST see:

- [Create, list, or delete a user-assigned managed identities using REST API calls](#)

# Configure a VM with managed identities for Azure resources using an Azure SDK

9/21/2022 • 2 minutes to read • [Edit Online](#)

Managed identities for Azure resources is a feature of Azure Active Directory. Each of the [Azure services that support managed identities for Azure resources](#) are subject to their own timeline. Make sure you review the [availability](#) status of managed identities for your resource and [known issues](#) before you begin.

Managed identities for Azure resources provide Azure services with an automatically managed identity in Azure Active Directory (AD). You can use this identity to authenticate to any service that supports Azure AD authentication, without having credentials in your code.

In this article, you learn how to enable and remove managed identities for Azure resources for an Azure VM, using an Azure SDK.

## Prerequisites

- If you're not familiar with the managed identities for Azure resources feature, see this [overview](#). If you don't have an Azure account, [sign up for a free account](#) before you continue.

## Azure SDKs with managed identities for Azure resources support

Azure supports multiple programming platforms through a series of [Azure SDKs](#). Several of them have been updated to support managed identities for Azure resources, and provide corresponding samples to demonstrate usage. This list is updated as other support is added:

SDK	SAMPLE
.NET	<a href="#">Manage resource from a VM enabled with managed identities for Azure resources enabled</a>
Java	<a href="#">Manage storage from a VM enabled with managed identities for Azure resources</a>
Node.js	<a href="#">Create a VM with system-assigned managed identity enabled</a>
Python	<a href="#">Create a VM with system-assigned managed identity enabled</a>
Ruby	<a href="#">Create Azure VM with a system-assigned identity enabled</a>

## Next steps

- See related articles under [Configure Identity for an Azure VM](#), to learn how you can also use the Azure portal, PowerShell, CLI, and resource templates.

# Log in to a Linux virtual machine in Azure by using Azure AD and OpenSSH

9/21/2022 • 21 minutes to read • [Edit Online](#)

To improve the security of Linux virtual machines (VMs) in Azure, you can integrate with Azure Active Directory (Azure AD) authentication. You can now use Azure AD as a core authentication platform and a certificate authority to SSH into a Linux VM by using Azure AD and OpenSSH certificate-based authentication. This functionality allows organizations to manage access to VMs with Azure role-based access control (RBAC) and Conditional Access policies.

This article shows you how to create and configure a Linux VM and log in with Azure AD by using OpenSSH certificate-based authentication.

## IMPORTANT

This capability is now generally available. The previous version that made use of device code flow was [deprecated on August 15, 2021](#). To migrate from the old version to this version, see the section [Migrate from the previous \(preview\) version](#).

There are many security benefits of using Azure AD with OpenSSH certificate-based authentication to log in to Linux VMs in Azure. They include:

- Use your Azure AD credentials to log in to Azure Linux VMs.
- Get SSH key-based authentication without needing to distribute SSH keys to users or provision SSH public keys on any Azure Linux VMs that you deploy. This experience is much simpler than having to worry about sprawl of stale SSH public keys that could cause unauthorized access.
- Reduce reliance on local administrator accounts, credential theft, and weak credentials.
- Help secure Linux VMs by configuring password complexity and password lifetime policies for Azure AD.
- With RBAC, specify who can log in to a VM as a regular user or with administrator privileges. When users join your team, you can update the Azure RBAC policy for the VM to grant access as appropriate. When employees leave your organization and their user accounts are disabled or removed from Azure AD, they no longer have access to your resources.
- With Conditional Access, configure policies to require multifactor authentication or to require that your client device is managed (for example, compliant or hybrid Azure AD joined) before you can use it SSH into Linux VMs.
- Use Azure deploy and audit policies to require Azure AD login for Linux VMs and flag unapproved local accounts.

Login to Linux VMs with Azure Active Directory works for customers who use Active Directory Federation Services.

## Supported Linux distributions and Azure regions

The following Linux distributions are currently supported for deployments in a supported region:

DISTRIBUTION	VERSION
Common Base Linux Mariner (CBL-Mariner)	CBL-Mariner 1, CBL-Mariner 2

DISTRIBUTION	VERSION
CentOS	CentOS 7, CentOS 8
Debian	Debian 9, Debian 10, Debian 11
openSUSE	openSUSE Leap 42.3, openSUSE Leap 15.1+
RedHat Enterprise Linux (RHEL)	RHEL 7.4 to RHEL 7.10, RHEL 8.3+
SUSE Linux Enterprise Server (SLES)	SLES 12, SLES 15.1+
Ubuntu Server	Ubuntu Server 16.04 to Ubuntu Server 22.04

The following Azure regions are currently supported for this feature:

- Azure Global
- Azure Government
- Azure China 21Vianet

Use of the SSH extension for Azure CLI on Azure Kubernetes Service (AKS) clusters is not supported. For more information, see [Support policies for AKS](#).

If you choose to install and use the Azure CLI locally, it must be version 2.22.1 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install the Azure CLI](#).

#### NOTE

This functionality is also available for [Azure Arc-enabled servers](#).

## Meet requirements for login with Azure AD using OpenSSH certificate-based authentication

To enable Azure AD login through SSH certificate-based authentication for Linux VMs in Azure, be sure to meet the following network, virtual machine, and client (SSH client) requirements.

### Network

VM network configuration must permit outbound access to the following endpoints over TCP port 443.

Azure Global:

- `https://packages.microsoft.com` : For package installation and upgrades.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.microsoftonline.com` : For PAM-based (pluggable authentication modules) authentication flows.
- `https://pas.windows.net` : For Azure RBAC flows.

Azure Government:

- `https://packages.microsoft.com` : For package installation and upgrades.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.microsoftonline.us` : For PAM-based authentication flows.
- `https://pasff.usgovcloudapi.net` : For Azure RBAC flows.

Azure China 21Vianet:

- `https://packages.microsoft.com` : For package installation and upgrades.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.chinacloudapi.cn` : For PAM-based authentication flows.
- `https://pas.chinacloudapi.cn` : For Azure RBAC flows.

## Virtual machine

Ensure that your VM is configured with the following functionality:

- System-assigned managed identity. This option is automatically selected when you use the Azure portal to create VMs and select the Azure AD login option. You can also enable system-assigned managed identity on a new or existing VM by using the Azure CLI.
- `aadsshlogin` and `aadsshlogin-selinux` (as appropriate). These packages are installed with the `AADSSHLoginForLinux` VM extension. The extension is installed when you use the Azure portal or the Azure CLI to create VMs and enable Azure AD login (**Management** tab).

## Client

Ensure that your client meets the following requirements:

- SSH client support for OpenSSH-based certificates for authentication. You can use Azure CLI (2.21.1 or later) with OpenSSH (included in Windows 10 version 1803 or later) or Azure Cloud Shell to meet this requirement.
- SSH extension for Azure CLI. You can install this extension by using `az extension add --name ssh`. You don't need to install this extension when you're using Azure Cloud Shell, because it comes preinstalled.

If you're using any SSH client other than the Azure CLI or Azure Cloud Shell that supports OpenSSH certificates, you'll still need to use the Azure CLI with the SSH extension to retrieve ephemeral SSH certificates and optionally a configuration file. You can then use the configuration file with your SSH client.

- TCP connectivity from the client to either the public or private IP address of the VM. (ProxyCommand or SSH forwarding to a machine with connectivity also works.)

### IMPORTANT

SSH clients based on PuTTY don't support OpenSSH certificates and can't be used to log in with Azure AD OpenSSH certificate-based authentication.

## Enable Azure AD login for a Linux VM in Azure

To use Azure AD login for a Linux VM in Azure, you need to first enable the Azure AD login option for your Linux VM. You then configure Azure role assignments for users who are authorized to log in to the VM. Finally, you use the SSH client that supports OpenSSH, such as the Azure CLI or Azure Cloud Shell, to SSH into your Linux VM.

There are two ways to enable Azure AD login for your Linux VM:

- The Azure portal experience when you're creating a Linux VM
- The Azure Cloud Shell experience when you're creating a Linux VM or using an existing one

### Azure portal

You can enable Azure AD login for any of the [supported Linux distributions](#) by using the Azure portal.

For example, to create an Ubuntu Server 18.04 Long Term Support (LTS) VM in Azure with Azure AD login:

1. Sign in to the Azure portal by using an account that has access to create VMs, and then select + **Create a**

resource.

2. Select **Create** under **Ubuntu Server 18.04 LTS** in the **Popular** view.
3. On the **Management** tab:
  - a. Select the **Login with Azure Active Directory** checkbox.
  - b. Ensure that the **System assigned managed identity** checkbox is selected.
4. Go through the rest of the experience of creating a virtual machine. You'll have to create an administrator account with username and password or SSH public key.

## Azure Cloud Shell

Azure Cloud Shell is a free, interactive shell that you can use to run the steps in this article. Common Azure tools are preinstalled and configured in Cloud Shell for you to use with your account. Just select the **Copy** button to copy the code, paste it in Cloud Shell, and then select the **Enter** key to run it.

There are a few ways to open Cloud Shell:

- Select **Try It** in the upper-right corner of a code block.
- Open Cloud Shell in your browser.
- Select the Cloud Shell button on the menu in the upper-right corner of the Azure portal.

If you choose to install and use the Azure CLI locally, this article requires you to use version 2.22.1 or later. Run `az --version` to find the version. If you need to install or upgrade, see [Install the Azure CLI](#).

1. Create a resource group by running [az group create](#).
2. Create a VM by running [az vm create](#). Use a supported distribution in a supported region.
3. Install the Azure AD login VM extension by using [az vm extension set](#).

The following example deploys a VM and then installs the extension to enable Azure AD login for a Linux VM. VM extensions are small applications that provide post-deployment configuration and automation tasks on Azure virtual machines. Customize the example as needed to support your testing requirements.

```
az group create --name AzureADLinuxVM --location southcentralus
az vm create \
    --resource-group AzureADLinuxVM \
    --name myVM \
    --image UbuntuLTS \
    --assign-identity \
    --admin-username azureuser \
    --generate-ssh-keys
az vm extension set \
    --publisher Microsoft.Azure.ActiveDirectory \
    --name AADSSHLoginForLinux \
    --resource-group AzureADLinuxVM \
    --vm-name myVM
```

It takes a few minutes to create the VM and supporting resources.

The AADSSHLoginForLinux extension can be installed on an existing (supported distribution) Linux VM with a running VM agent to enable Azure AD authentication. If you're deploying this extension to a previously created VM, the VM must have at least 1 GB of memory allocated or the installation will fail.

The `provisioningState` value of `Succeeded` appears when the extension is successfully installed on the VM. The VM must have a running [VM agent](#) to install the extension.

## Configure role assignments for the VM

Now that you've created the VM, you need to configure an Azure RBAC policy to determine who can log in to the VM. Two Azure roles are used to authorize VM login:

- **Virtual Machine Administrator Login:** Users who have this role assigned can log in to an Azure virtual machine with administrator privileges.
- **Virtual Machine User Login:** Users who have this role assigned can log in to an Azure virtual machine with regular user privileges.

To allow a user to log in to a VM over SSH, you must assign the Virtual Machine Administrator Login or Virtual Machine User Login role on the resource group that contains the VM and its associated virtual network, network interface, public IP address, or load balancer resources.

An Azure user who has the Owner or Contributor role assigned for a VM doesn't automatically have privileges to Azure AD login to the VM over SSH. There's an intentional (and audited) separation between the set of people who control virtual machines and the set of people who can access virtual machines.

There are two ways to configure role assignments for a VM:

- Azure AD portal experience
- Azure Cloud Shell experience

#### NOTE

The Virtual Machine Administrator Login and Virtual Machine User Login roles use `dataActions` and can be assigned at the management group, subscription, resource group, or resource scope. We recommend that you assign the roles at the management group, subscription, or resource level and not at the individual VM level. This practice avoids the risk of reaching the [Azure role assignments limit](#) per subscription.

## Azure AD portal

To configure role assignments for your Azure AD-enabled Linux VMs:

1. For **Resource Group**, select the resource group that contains the VM and its associated virtual network, network interface, public IP address, or load balancer resource.
2. Select **Access control (IAM)**.
3. Select **Add > Add role assignment** to open the **Add role assignment** page.
4. Assign the following role. For detailed steps, see [Assign Azure roles by using the Azure portal](#).

SETTING	VALUE
Role	Virtual Machine Administrator Login or Virtual Machine User Login
Assign access to	User, group, service principal, or managed identity

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Name ↑↓	Description ↑↓	Type ↑↓	Category ↑↓	Details
Owner	Grants full access to manage all resources, including the ability to a...	BuiltinRole	General	<a href="#">View</a>
Contributor	Grants full access to manage all resources, but does not allow you ...	BuiltinRole	General	<a href="#">View</a>
Reader	View all resources, but does not allow you to make any changes.	BuiltinRole	General	<a href="#">View</a>
AcrDelete	acr delete	BuiltinRole	Containers	<a href="#">View</a>
AcrImageSigner	acr image signer	BuiltinRole	Containers	<a href="#">View</a>
AcrPull	acr pull	BuiltinRole	Containers	<a href="#">View</a>
AcrPush	acr push	BuiltinRole	Containers	<a href="#">View</a>
AcrQuarantineReader	acr quarantine data reader	BuiltinRole	Containers	<a href="#">View</a>
AcrQuarantineWriter	acr quarantine data writer	BuiltinRole	Containers	<a href="#">View</a>

After a few moments, the security principal is assigned the role at the selected scope.

## Azure Cloud Shell

The following example uses `az role assignment create` to assign the Virtual Machine Administrator Login role to the VM for your current Azure user. You obtain the username of your current Azure account by using `az account show`, and you set the scope to the VM created in a previous step by using `az vm show`.

You can also assign the scope at a resource group or subscription level. Normal Azure RBAC inheritance permissions apply.

```
username=$(az account show --query user.name --output tsv)
rg=$(az group show --resource-group myResourceGroup --query id -o tsv)

az role assignment create \
    --role "Virtual Machine Administrator Login" \
    --assignee $username \
    --scope $rg
```

### NOTE

If your Azure AD domain and login username domain don't match, you must specify the object ID of your user account by using `--assignee-object-id`, not just the username for `--assignee`. You can obtain the object ID for your user account by using `az ad user list`.

For more information on how to use Azure RBAC to manage access to your Azure subscription resources, see [Steps to assign an Azure role](#).

## Install the SSH extension for Azure CLI

If you're using Azure Cloud Shell, no other setup is needed because both the minimum required version of the Azure CLI and the SSH extension for Azure CLI are already included in the Cloud Shell environment.

Run the following command to add the SSH extension for Azure CLI:

```
az extension add --name ssh
```

The minimum version required for the extension is 0.1.4. Check the installed version by using the following command:

```
az extension show --name ssh
```

## Enforce Conditional Access policies

You can enforce Conditional Access policies that are enabled with Azure AD login, such as:

- Requiring multifactor authentication.
- Requiring a compliant or hybrid Azure AD-joined device for the device running the SSH client.
- Checking for risks before authorizing access to Linux VMs in Azure.

The application that appears in the Conditional Access policy is called *Azure Linux VM Sign-In*.

### NOTE

Conditional Access policy enforcement that requires device compliance or hybrid Azure AD join on the device that's running the SSH client works only with the Azure CLI that's running on Windows and macOS. It's not supported when you're using the Azure CLI on Linux or Azure Cloud Shell.

### Missing application

If the Azure Linux VM Sign-In application is missing from Conditional Access, make sure the application isn't in the tenant:

1. Sign in to the Azure portal.
2. Browse to **Azure Active Directory > Enterprise applications**.
3. Remove the filters to see all applications, and search for **VM**. If you don't see Azure Linux VM Sign-In as a result, the service principal is missing from the tenant.

Another way to verify it is via Graph PowerShell:

1. [Install the Graph PowerShell SDK](#) if you haven't already done so.
2. Enter the command  

```
Connect-MgGraph -Scopes "ServicePrincipalEndpoint.ReadWrite.All","Application.ReadWrite.All"
```
3. Sign in with a Global Admin account.
4. Consent to the prompt that asks for your permission.
5. Enter the command  

```
Get-MgServicePrincipal -ConsistencyLevel eventual -Search '"DisplayName:Azure Linux VM Sign-In"'
```

If this command results in no output and returns you to the PowerShell prompt, you can create the service principal by using the following Graph PowerShell command:

```
New-MgServicePrincipal -AppId ce6ff14a-7fdc-4685-bbe0-f6afdfcfa8e0
```

Successful output will show that the app ID and the application name Azure Linux VM Sign-In were created.

6. Sign out of Graph PowerShell by using the following command: `Disconnect-MgGraph`.

# Log in by using an Azure AD user account to SSH into the Linux VM

## Log in by using the Azure CLI

Enter `az login`. This command opens a browser window, where you can sign in by using your Azure AD account.

```
az login
```

Then enter `az ssh vm`. The following example automatically resolves the appropriate IP address for the VM.

```
az ssh vm -n myVM -g AzureADLinuxVM
```

If you're prompted, enter your Azure AD login credentials at the login page, perform multifactor authentication, and/or satisfy device checks. You'll be prompted only if your Azure CLI session doesn't already meet any required Conditional Access criteria. Close the browser window, return to the SSH prompt, and you'll be automatically connected to the VM.

You're now signed in to the Linux virtual machine with the role permissions as assigned, such as VM User or VM Administrator. If your user account is assigned the Virtual Machine Administrator Login role, you can use sudo to run commands that require root privileges.

## Log in by using Azure Cloud Shell

You can use Azure Cloud Shell to connect to VMs without needing to install anything locally to your client machine. Start Cloud Shell by selecting the shell icon in the upper-right corner of the Azure portal.

Cloud Shell automatically connects to a session in the context of the signed-in user. Now run `az login` again and go through the interactive sign-in flow:

```
az login
```

Then you can use the normal `az ssh vm` commands to connect by using the name and resource group or IP address of the VM:

```
az ssh vm -n myVM -g AzureADLinuxVM
```

### NOTE

Conditional Access policy enforcement that requires device compliance or hybrid Azure AD join is not supported when you're using Azure Cloud Shell.

# Log in by using the Azure AD service principal to SSH into the Linux VM

The Azure CLI supports authenticating with a service principal instead of a user account. Because service principals aren't tied to any particular user, customers can use them to SSH into a VM to support any automation scenarios they might have. The service principal must have VM Administrator or VM User rights assigned. Assign permissions at the subscription or resource group level.

The following example will assign VM Administrator rights to the service principal at the resource group level. Replace the placeholders for service principal object ID, subscription ID, and resource group name.

```
az role assignment create \
--role "Virtual Machine Administrator Login" \
--assignee-object-id <service-principal-objectid> \
--assignee-principal-type ServicePrincipal \
--scope "/subscriptions/<subscription-id>/resourceGroups/<resourcegroup-name>"
```

Use the following example to authenticate to the Azure CLI by using the service principal. For more information, see the article [Sign in to the Azure CLI with a service principal](#).

```
az login --service-principal -u <sp-app-id> -p <password-or-cert> --tenant <tenant-id>
```

When authentication with a service principal is complete, use the normal Azure CLI SSH commands to connect to the VM:

```
az ssh vm -n myVM -g AzureADLinuxVM
```

## Export the SSH configuration for use with SSH clients that support OpenSSH

Login to Azure Linux VMs with Azure AD supports exporting the OpenSSH certificate and configuration. That means you can use any SSH clients that support OpenSSH-based certificates to sign in through Azure AD. The following example exports the configuration for all IP addresses assigned to the VM:

```
az ssh config --file ~/.ssh/config -n myVM -g AzureADLinuxVM
```

Alternatively, you can export the configuration by specifying just the IP address. Replace the IP address in the following example with the public or private IP address for your VM. (You must bring your own connectivity for private IPs.) Enter `az ssh config -h` for help with this command.

```
az ssh config --file ~/.ssh/config --ip 10.11.123.456
```

You can then connect to the VM through normal OpenSSH usage. Connection can be done through any SSH client that uses OpenSSH.

## Run sudo with Azure AD login

After users who are assigned the VM Administrator role successfully SSH into a Linux VM, they'll be able to run sudo with no other interaction or authentication requirement. Users who are assigned the VM User role won't be able to run sudo.

## Connect to VMs in virtual machine scale sets

Virtual machine scale sets are supported, but the steps are slightly different for enabling and connecting to VMs in a virtual machine scale set:

1. Create a virtual machine scale set or choose one that already exists. Enable a system-assigned managed identity for your virtual machine scale set:

```
az vmss identity assign --name myVMSS --resource-group AzureADLinuxVM
```

## 2. Install the Azure AD extension on your virtual machine scale set:

```
az vmss extension set --publisher Microsoft.Azure.ActiveDirectory --name AADSSHLoginForLinux --resource-group AzureADLinuxVM --vmss-name myVMSS
```

Virtual machine scale sets usually don't have public IP addresses. You must have connectivity to them from another machine that can reach their Azure virtual network. This example shows how to use the private IP of a VM in a virtual machine scale set to connect from a machine in the same virtual network:

```
az ssh vm --ip 10.11.123.456
```

### NOTE

You can't automatically determine the virtual machine scale set VM's IP addresses by using the `--resource-group` and `--name` switches.

## Migrate from the previous (preview) version

If you're using the previous version of Azure AD login for Linux that was based on device code flow, complete the following steps by using the Azure CLI:

### 1. Uninstall the AADLoginForLinux extension on the VM:

```
az vm extension delete -g MyResourceGroup --vm-name MyVm -n AADLoginForLinux
```

### NOTE

Uninstallation of the extension can fail if there are any Azure AD users currently logged in on the VM. Make sure all users are logged out first.

### 2. Enable system-assigned managed identity on your VM:

```
az vm identity assign -g myResourceGroup -n myVm
```

### 3. Install the AADSSHLoginForLinux extension on the VM:

```
az vm extension set \
--publisher Microsoft.Azure.ActiveDirectory \
--name AADSSHLoginForLinux \
--resource-group myResourceGroup \
--vm-name myVm
```

## Use Azure Policy to meet standards and assess compliance

Use Azure Policy to:

- Ensure that Azure AD login is enabled for your new and existing Linux virtual machines.
- Assess compliance of your environment at scale on a compliance dashboard.

With this capability, you can use many levels of enforcement. You can flag new and existing Linux VMs within your environment that don't have Azure AD login enabled. You can also use Azure Policy to deploy the Azure AD

extension on new Linux VMs that don't have Azure AD login enabled, as well as remediate existing Linux VMs to the same standard.

In addition to these capabilities, you can use Azure Policy to detect and flag Linux VMs that have unapproved local accounts created on their machines. To learn more, review [Azure Policy](#).

## Troubleshoot sign-in issues

Use the following sections to correct common errors that can happen when you try to SSH with Azure AD credentials.

### Couldn't retrieve token from local cache

If you get a message that says the token couldn't be retrieved from the local cache, you must run `az login` again and go through an interactive sign-in flow. Review the section about [logging in by using Azure Cloud Shell](#).

### Access denied: Azure role not assigned

If you see an "Azure role not assigned" error on your SSH prompt, verify that you've configured Azure RBAC policies for the VM that grants the user either the Virtual Machine Administrator Login role or the Virtual Machine User Login role. If you're having problems with Azure role assignments, see the article [Troubleshoot Azure RBAC](#).

### Problems deleting the old (AADLoginForLinux) extension

If the uninstallation scripts fail, the extension might get stuck in a transitioning state. When this happens, the extension can leave packages that it's supposed to uninstall during its removal. In such cases, it's better to manually uninstall the old packages and then try to run the `az vm extension delete` command.

To uninstall old packages:

1. Log in as a local user with admin privileges.
2. Make sure there are no logged-in Azure AD users. Call the `who -u` command to see who is logged in. Then use `sudo kill <pid>` for all session processes that the previous command reported.
3. Run `sudo apt remove --purge aadlogin` (Ubuntu/Debian), `sudo yum erase aadlogin` (RHEL or CentOS), or `sudo zypper remove aadlogin` (openSUSE or SLES).
4. If the command fails, try the low-level tools with scripts disabled:
  - a. For Ubuntu/Debian, run `sudo dpkg --purge aadlogin`. If it's still failing because of the script, delete the `/var/lib/dpkg/info/aadlogin.prerm` file and try again.
  - b. For everything else, run `rpm -e --noscripts aadlogin`.
5. Repeat steps 3-4 for package `aadlogin-selinux`.

### Extension installation errors

Installation of the AADSSHLoginForLinux VM extension to existing computers might fail with one of the following known error codes.

#### Non-zero exit code 22

If you get exit code 22, the status of the AADSSHLoginForLinux VM extension shows as **Transitioning** in the portal.

This failure happens because a system-assigned managed identity is required.

The solution is to:

1. Uninstall the failed extension.
2. Enable a system-assigned managed identity on the Azure VM.
3. Run the extension installation command again.

## Non-zero exit code 23

If you get exit code 23, the status of the AADSSHLoginForLinux VM extension shows as **Transitioning** in the portal.

This failure happens when the older AADLoginForLinux VM extension is still installed.

The solution is to uninstall the older AADLoginForLinux VM extension from the VM. The status of the new AADSSHLoginForLinux VM extension will then change to **Provisioning succeeded** in the portal.

## The az ssh vm command fails with KeyError access\_token

If the `az ssh vm` command fails, you're using an outdated version of the Azure CLI client.

The solution is to upgrade the Azure CLI client to version 2.21.0 or later.

## SSH connection is closed

After a user successfully signs in by using `az login`, connection to the VM through `az ssh vm -ip <address>` or `az ssh vm --name <vm_name> -g <resource_group>` might fail with "Connection closed by <ip\_address> port 22."

One cause for this error is that the user isn't assigned to the Virtual Machine Administrator Login or Virtual Machine User Login role within the scope of this VM. In that case, the solution is to add the user to one of those Azure RBAC roles within the scope of this VM.

This error can also happen if the user is in a required Azure RBAC role, but the system-assigned managed identity has been disabled on the VM. In that case, perform these actions:

1. Enable the system-assigned managed identity on the VM.
2. Allow several minutes to pass before the user tries to connect by using `az ssh vm --ip <ip_address>`.

## Connection problems with virtual machine scale sets

VM connections with virtual machine scale sets can fail if the scale set instances are running an old model.

Upgrading scale set instances to the latest model might resolve the problem, especially if an upgrade hasn't been done since the Azure AD Login extension was installed. Upgrading an instance applies a standard scale set configuration to the individual instance.

## AllowGroups or DenyGroups statements in sshd\_config cause the first login to fail for Azure AD users

If `sshd_config` contains either `AllowGroups` or `DenyGroups` statements, the first login fails for Azure AD users. If the statement was added after users have already had a successful login, they can log in.

One solution is to remove `AllowGroups` and `DenyGroups` statements from `sshd_config`.

Another solution is to move `AllowGroups` and `DenyGroups` to a `match user` section in `sshd_config`. Make sure the match template excludes Azure AD users.

## Next steps

- [What is a device identity?](#)
- [Common Conditional Access policies](#)

# Log in to a Windows virtual machine in Azure by using Azure AD

9/21/2022 • 19 minutes to read • [Edit Online](#)

Organizations can improve the security of Windows virtual machines (VMs) in Azure by integrating with Azure Active Directory (Azure AD) authentication. You can now use Azure AD as a core authentication platform to RDP into *Windows Server 2019 Datacenter edition* and later, or *Windows 10 1809* and later. You can then centrally control and enforce Azure role-based access control (RBAC) and Conditional Access policies that allow or deny access to the VMs.

This article shows you how to create and configure a Windows VM and log in by using Azure AD-based authentication.

There are many security benefits of using Azure AD-based authentication to log in to Windows VMs in Azure. They include:

- Use Azure AD credentials to log in to Windows VMs in Azure. The result is federated and managed domain users.
- Reduce reliance on local administrator accounts.
- Password complexity and password lifetime policies that you configure for Azure AD also help secure Windows VMs.
- With Azure RBAC:
  - Specify who can log in to a VM as a regular user or with administrator privileges.
  - When users join or leave your team, you can update the Azure RBAC policy for the VM to grant access as appropriate.
  - When employees leave your organization and their user accounts are disabled or removed from Azure AD, they no longer have access to your resources.
- Configure Conditional Access policies to require multifactor authentication (MFA) and other signals, such as user sign-in risk, before you can RDP into Windows VMs.
- Use Azure deploy and audit policies to require Azure AD login for Windows VMs and to flag the use of unapproved local accounts on the VMs.
- Use Intune to automate and scale Azure AD join with mobile device management (MDM) auto-enrollment of Azure Windows VMs that are part of your virtual desktop infrastructure (VDI) deployments.

MDM auto-enrollment requires Azure AD Premium P1 licenses. Windows Server VMs don't support MDM enrollment.

## NOTE

After you enable this capability, your Windows VMs in Azure will be Azure AD joined. You cannot join them to another domain, like on-premises Active Directory or Azure Active Directory Domain Services. If you need to do so, disconnect the VM from Azure AD by uninstalling the extension.

## Requirements

## Supported Azure regions and Windows distributions

This feature currently supports the following Windows distributions:

- Windows Server 2019 Datacenter and later
- Windows 10 1809 and later

### IMPORTANT

Remote connection to VMs that are joined to Azure AD is allowed only from Windows 10 or later PCs that are Azure AD registered (starting with Windows 10 20H1), Azure AD joined, or hybrid Azure AD joined to the *same* directory as the VM.

This feature is now available in the following Azure clouds:

- Azure Global
- Azure Government
- Azure China 21Vianet

## Network requirements

To enable Azure AD authentication for your Windows VMs in Azure, you need to ensure that your VM's network configuration permits outbound access to the following endpoints over TCP port 443.

Azure Global:

- `https://enterpriseregistration.windows.net` : For device registration.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.microsoftonline.com` : For authentication flows.
- `https://pas.windows.net` : For Azure RBAC flows.

Azure Government:

- `https://enterpriseregistration.microsoftonline.us` : For device registration.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.microsoftonline.us` : For authentication flows.
- `https://pasff.usgovcloudapi.net` : For Azure RBAC flows.

Azure China 21Vianet:

- `https://enterpriseregistration.partner.microsoftonline.cn` : For device registration.
- `http://169.254.169.254` : Azure Instance Metadata Service endpoint.
- `https://login.chinacloudapi.cn` : For authentication flows.
- `https://pas.chinacloudapi.cn` : For Azure RBAC flows.

## Enable Azure AD login for a Windows VM in Azure

To use Azure AD login for a Windows VM in Azure, you must:

1. Enable the Azure AD login option for the VM.
2. Configure Azure role assignments for users who are authorized to log in to the VM.

There are two ways to enable Azure AD login for your Windows VM:

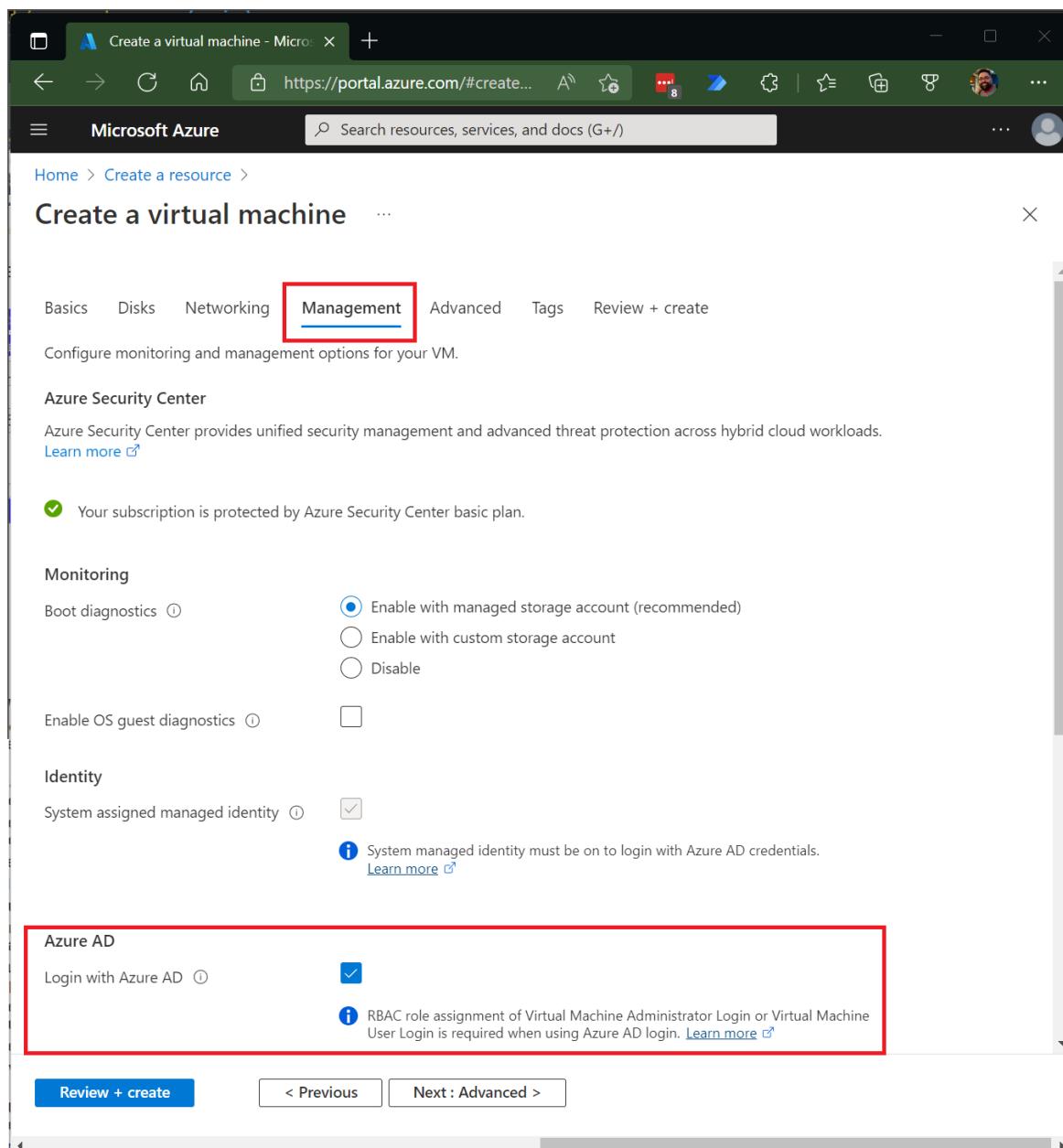
- The Azure portal, when you're creating a Windows VM.
- Azure Cloud Shell, when you're creating a Windows VM or using an existing Windows VM.

## Azure portal

You can enable Azure AD login for VM images in Windows Server 2019 Datacenter or Windows 10 1809 and later.

To create a Windows Server 2019 Datacenter VM in Azure with Azure AD login:

1. Sign in to the [Azure portal](#) by using an account that has access to create VMs, and select + **Create a resource**.
2. In the **Search the Marketplace** search bar, type **Windows Server**.
3. Select **Windows Server**, and then choose **Windows Server 2019 Datacenter** from the **Select a software plan** dropdown list.
4. Select **Create**.
5. On the **Management** tab, select the **Login with Azure AD** checkbox in the **Azure AD** section.



6. Make sure that **System assigned managed identity** in the **Identity** section is selected. This action should happen automatically after you enable login with Azure AD.
7. Go through the rest of the experience of creating a virtual machine. You'll have to create an administrator username and password for the VM.

## NOTE

To log in to the VM by using your Azure AD credentials, you first need to [configure role assignments](#) for the VM.

## Azure Cloud Shell

Azure Cloud Shell is a free, interactive shell that you can use to run the steps in this article. Common Azure tools are preinstalled and configured in Cloud Shell for you to use with your account. Just select the **Copy** button to copy the code, paste it in Cloud Shell, and then select the Enter key to run it. There are a few ways to open Cloud Shell:

- Select **Try It** in the upper-right corner of a code block.
- Open Cloud Shell in your browser.
- Select the Cloud Shell button on the menu in the upper-right corner of the [Azure portal](#).

This article requires you to run Azure CLI version 2.0.31 or later. Run `az --version` to find the version. If you need to install or upgrade, see the article [Install the Azure CLI](#).

1. Create a resource group by running `az group create`.
2. Create a VM by running `az vm create`. Use a supported distribution in a supported region.
3. Install the Azure AD login VM extension.

The following example deploys a VM named `myVM` (that uses `Win2019Datacenter`) into a resource group named `myResourceGroup`, in the `southcentralus` region. In this example and the next one, you can provide your own resource group and VM names as needed.

```
az group create --name myResourceGroup --location southcentralus

az vm create \
    --resource-group myResourceGroup \
    --name myVM \
    --image Win2019Datacenter \
    --assign-identity \
    --admin-username azureuser \
    --admin-password yourpassword
```

## NOTE

You must enable system-assigned managed identity on your virtual machine before you install the Azure AD login VM extension.

It takes a few minutes to create the VM and supporting resources.

Finally, install the Azure AD login VM extension to enable Azure AD login for Windows VMs. VM extensions are small applications that provide post-deployment configuration and automation tasks on Azure virtual machines. Use `az vm extension` set to install the `AADLoginForWindows` extension on the VM named `myVM` in the `myResourceGroup` resource group.

You can install the `AADLoginForWindows` extension on an existing Windows Server 2019 or Windows 10 1809 and later VM to enable it for Azure AD authentication. The following example uses the Azure CLI to install the extension:

```
az vm extension set \
--publisher Microsoft.Azure.ActiveDirectory \
--name AADLoginForWindows \
--resource-group myResourceGroup \
--vm-name myVM
```

After the extension is installed on the VM, `provisioningState` shows `Succeeded`.

## Configure role assignments for the VM

Now that you've created the VM, you need to configure an Azure RBAC policy to determine who can log in to the VM. Two Azure roles are used to authorize VM login:

- **Virtual Machine Administrator Login:** Users who have this role assigned can log in to an Azure virtual machine with administrator privileges.
- **Virtual Machine User Login:** Users who have this role assigned can log in to an Azure virtual machine with regular user privileges.

To allow a user to log in to the VM over RDP, you must assign the Virtual Machine Administrator Login or Virtual Machine User Login role to the resource group that contains the VM and its associated virtual network, network interface, public IP address, or load balancer resources.

An Azure user who has the Owner or Contributor role assigned for a VM does not automatically have privileges to log in to the VM over RDP. The reason is to provide audited separation between the set of people who control virtual machines and the set of people who can access virtual machines.

There are two ways to configure role assignments for a VM:

- Azure AD portal experience
- Azure Cloud Shell experience

### NOTE

The Virtual Machine Administrator Login and Virtual Machine User Login roles use `dataActions`, so they can't be assigned at the management group scope. Currently, you can assign these roles only at the subscription, resource group, or resource scope.

### Azure AD portal

To configure role assignments for your Azure AD-enabled Windows Server 2019 Datacenter VMs:

1. For **Resource Group**, select the resource group that contains the VM and its associated virtual network, network interface, public IP address, or load balancer resource.
2. Select **Access control (IAM)**.
3. Select **Add > Add role assignment** to open the **Add role assignment** page.
4. Assign the following role. For detailed steps, see [Assign Azure roles by using the Azure portal](#).

SETTING	VALUE
Role	Virtual Machine Administrator Login or Virtual Machine User Login
Assign access to	User, group, service principal, or managed identity

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. [Learn more](#)

Name ↑↓	Description ↑↓	Type ↑↓	Category ↑↓	Details
Owner	Grants full access to manage all resources, including the ability to a...	BuiltinRole	General	<a href="#">View</a>
Contributor	Grants full access to manage all resources, but does not allow you ...	BuiltinRole	General	<a href="#">View</a>
Reader	View all resources, but does not allow you to make any changes.	BuiltinRole	General	<a href="#">View</a>
AcrDelete	acr delete	BuiltinRole	Containers	<a href="#">View</a>
AcrImageSigner	acr image signer	BuiltinRole	Containers	<a href="#">View</a>
AcrPull	acr pull	BuiltinRole	Containers	<a href="#">View</a>
AcrPush	acr push	BuiltinRole	Containers	<a href="#">View</a>
AcrQuarantineReader	acr quarantine data reader	BuiltinRole	Containers	<a href="#">View</a>
AcrQuarantineWriter	acr quarantine data writer	BuiltinRole	Containers	<a href="#">View</a>

## Azure Cloud Shell

The following example uses [az role assignment create](#) to assign the Virtual Machine Administrator Login role to the VM for your current Azure user. You obtain the username of your current Azure account by using [az account show](#), and you set the scope to the VM created in a previous step by using [az vm show](#).

You can also assign the scope at a resource group or subscription level. Normal Azure RBAC inheritance permissions apply. For more information, see [Log in to a Linux virtual machine in Azure by using Azure Active Directory authentication](#).

```
$username=$(az account show --query user.name --output tsv)
$rg=$(az group show --resource-group myResourceGroup --query id -o tsv)

az role assignment create \
    --role "Virtual Machine Administrator Login" \
    --assignee $username \
    --scope $rg
```

### NOTE

If your Azure AD domain and login username domain don't match, you must specify the object ID of your user account by using `--assignee-object-id`, not just the username for `--assignee`. You can obtain the object ID for your user account by using [az ad user list](#).

For more information about how to use Azure RBAC to manage access to your Azure subscription resources, see the following articles:

- [Assign Azure roles by using the Azure CLI](#)
- [Assign Azure roles by using the Azure portal](#)
- [Assign Azure roles by using Azure PowerShell](#)

## Enforce Conditional Access policies

You can enforce Conditional Access policies, such as multifactor authentication or user sign-in risk check, before

you authorize access to Windows VMs in Azure that are enabled with Azure AD login. To apply a Conditional Access policy, you must select the **Azure Windows VM Sign-In** app from the cloud apps or actions assignment option. Then use sign-in risk as a condition and/or require MFA as a control for granting access.

#### NOTE

If you require MFA as a control for granting access to the Azure Windows VM Sign-In app, then you must supply an MFA claim as part of the client that initiates the RDP session to the target Windows VM in Azure. The only way to achieve this on a Windows 10 or later client is to use a Windows Hello for Business PIN or biometric authentication with the RDP client. Support for biometric authentication was added to the RDP client in Windows 10 version 1809.

Remote desktop using Windows Hello for Business authentication is available only for deployments that use a certificate trust model. It's currently not available for a key trust model.

## Log in by using Azure AD credentials to a Windows VM

#### IMPORTANT

Remote connection to VMs that are joined to Azure AD is allowed only from Windows 10 or later PCs that are either Azure AD registered (minimum required build is 20H1) or Azure AD joined or hybrid Azure AD joined to the *same* directory as the VM. Additionally, to RDP by using Azure AD credentials, users must belong to one of the two Azure roles, Virtual Machine Administrator Login or Virtual Machine User Login.

If you're using an Azure AD-registered Windows 10 or later PC, you must enter credentials in the `AzureAD\UPN` format (for example, `AzureAD\john@contoso.com`). At this time, you can use Azure Bastion to log in with Azure AD authentication [via the Azure CLI and the native RDP client mstsc](#).

To log in to your Windows Server 2019 virtual machine by using Azure AD:

1. Go to the overview page of the virtual machine that has been enabled with Azure AD login.
2. Select **Connect** to open the **Connect to virtual machine** pane.
3. Select **Download RDP File**.
4. Select **Open** to open the Remote Desktop Connection client.
5. Select **Connect** to open the Windows login dialog.
6. Log in by using your Azure AD credentials.

You're now logged in to the Windows Server 2019 Azure virtual machine with the role permissions as assigned, such as VM User or VM Administrator.

#### NOTE

You can save the .rdp file locally on your computer to start future remote desktop connections to your virtual machine, instead of going to the virtual machine overview page in the Azure portal and using the connect option.

## Use Azure Policy to meet standards and assess compliance

Use Azure Policy to:

- Ensure that Azure AD login is enabled for your new and existing Windows virtual machines.
- Assess compliance of your environment at scale on a compliance dashboard.

With this capability, you can use many levels of enforcement. You can flag new and existing Windows VMs within your environment that don't have Azure AD login enabled. You can also use Azure Policy to deploy the Azure AD extension on new Windows VMs that don't have Azure AD login enabled, and remediate existing

Windows VMs to the same standard.

In addition to these capabilities, you can use Azure Policy to detect and flag Windows VMs that have unapproved local accounts created on their machines. To learn more, review [Azure Policy](#).

## Troubleshoot deployment problems

The AADLoginForWindows extension must be installed successfully for the VM to complete the Azure AD join process. If the VM extension fails to be installed correctly, perform the following steps:

1. RDP to the VM by using the local administrator account and examine the *CommandExecution.log* file under *C:\WindowsAzure\Logs\Plugins\Microsoft.Azure.ActiveDirectoryAADLoginForWindows\1.0.0.1*.

### NOTE

If the extension restarts after the initial failure, the log with the deployment error will be saved as *CommandExecution\_YYYYMMDDHHMMSSSS.log*.

2. Open a PowerShell window on the VM. Verify that the following queries against the Azure Instance Metadata Service endpoint running on the Azure host return the expected output:

COMMAND TO RUN	EXPECTED OUTPUT
<pre>curl -H Metadata:true "http://169.254.169.254/metadata/instance?api-version=2017-08-01"</pre>	Correct information about the Azure VM
<pre>curl -H Metadata:true "http://169.254.169.254/metadata/identity/info?api-version=2018-02-01"</pre>	Valid tenant ID associated with the Azure subscription
<pre>curl -H Metadata:true "http://169.254.169.254/metadata/identity/oauth2/token?resource=urn:ms-drivers:enterpriseregistration.windows.net&amp;api-version=2018-02-01"</pre>	Valid access token issued by Azure Active Directory for the managed identity that is assigned to this VM

### NOTE

You can decode the access token by using a tool like [calebb.net](#). Verify that the `oid` value in the access token matches the managed identity that's assigned to the VM.

3. Ensure that the required endpoints are accessible from the VM via PowerShell:

- curl.exe https://login.microsoftonline.com/ -D -
- curl.exe https://login.microsoftonline.com/<TenantID>/ -D -
- curl.exe https://enterpriseregistration.windows.net/ -D -
- curl.exe https://device.login.microsoftonline.com/ -D -
- curl.exe https://pas.windows.net/ -D -

### NOTE

Replace `<TenantID>` with the Azure AD tenant ID that's associated with the Azure subscription.

`login.microsoftonline.com/<TenantID>`, `enterpriseregistration.windows.net`, and `pas.windows.net` should return 404 Not Found, which is expected behavior.

4. View the device state by running `dsregcmd /status`. The goal is for the device state to show as

`AzureAdJoined : YES`.

#### NOTE

Azure AD join activity is captured in Event Viewer under the *User Device Registration\Admin* log at *Event Viewer (local)\Applications and Services Logs\Windows\Microsoft\User Device Registration\Admin*.

If the AADLoginForWindows extension fails with an error code, you can perform the following steps.

#### Terminal error code 1007 and exit code -2145648574.

Terminal error code 1007 and exit code -2145648574 translate to `DSREG_E_MSI_TENANTID_UNAVAILABLE`. The extension can't query the Azure AD tenant information.

Connect to the VM as a local administrator and verify that the endpoint returns a valid tenant ID from Azure Instance Metadata Service. Run the following command from an elevated PowerShell window on the VM:

```
curl -H Metadata:true http://169.254.169.254/metadata/identity/info?api-version=2018-02-01
```

This problem can also happen when the VM admin attempts to install the AADLoginForWindows extension, but a system-assigned managed identity hasn't enabled the VM first. In that case, go to the **Identity** pane of the VM. On the **System assigned** tab, verify that the **Status** toggle is set to **On**.

#### Exit code -2145648607

Exit code -2145648607 translates to `DSREG_AUTOJOIN_DISC_FAILED`. The extension can't reach the `https://enterpriseregistration.windows.net` endpoint.

1. Verify that the required endpoints are accessible from the VM via PowerShell:

- `curl https://login.microsoftonline.com/ -D -`
- `curl https://login.microsoftonline.com/<TenantID>/ -D -`
- `curl https://enterpriseregistration.windows.net/ -D -`
- `curl https://device.login.microsoftonline.com/ -D -`
- `curl https://pas.windows.net/ -D -`

#### NOTE

Replace `<TenantID>` with the Azure AD tenant ID that's associated with the Azure subscription. If you need to find the tenant ID, you can hover over your account name or select **Azure Active Directory > Properties > Directory ID** in the Azure portal.

Attempts to connect to `enterpriseregistration.windows.net` might return 404 Not Found, which is expected behavior. Attempts to connect to `pas.windows.net` might prompt for PIN credentials or might return 404 Not Found. (You don't need to enter the PIN.) Either one is sufficient to verify that the URL is reachable.

2. If any of the commands fails with "Could not resolve host `<URL>`," try running this command to determine which DNS server the VM is using:

```
nslookup <URL>
```

#### NOTE

Replace `<URL>` with the fully qualified domain names that the endpoints use, such as `login.microsoftonline.com`.

3. See whether specifying a public DNS server allows the command to succeed:

```
nslookup <URL> 208.67.222.222
```

4. If necessary, change the DNS server that's assigned to the network security group that the Azure VM belongs to.

### Exit code 51

Exit code 51 translates to "This extension is not supported on the VM's operating system."

The AADLoginForWindows extension is intended to be installed only on Windows Server 2019 or Windows 10 (Build 1809 or later). Ensure that your version or build of Windows is supported. If it isn't supported, uninstall the extension.

## Troubleshoot sign-in problems

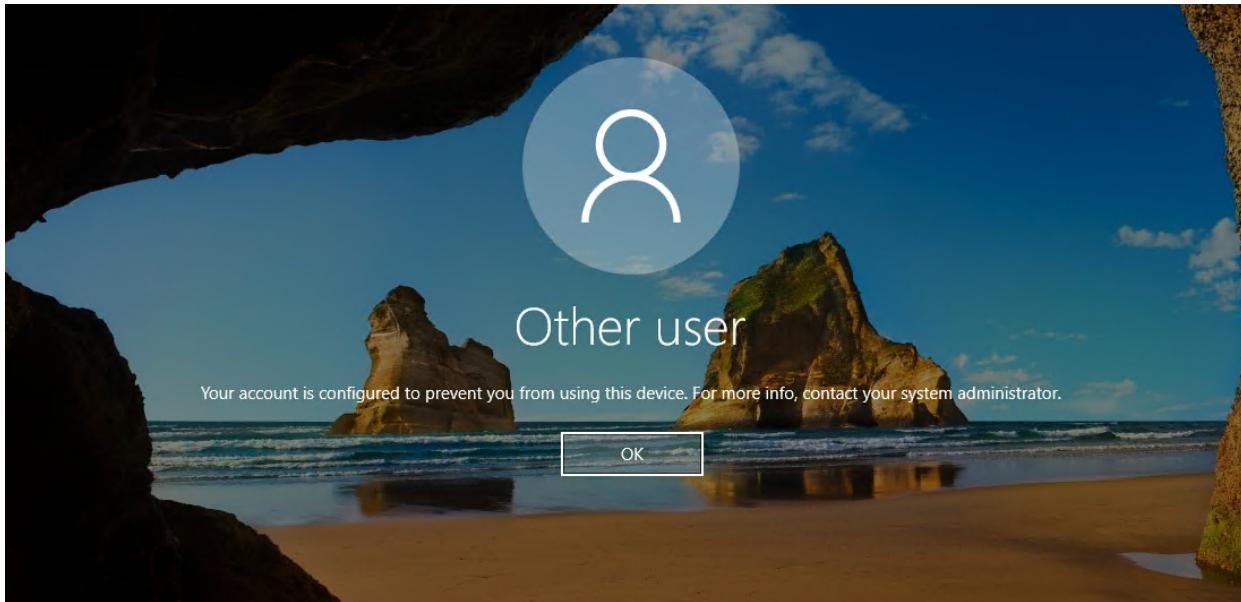
Use the following information to correct sign-in problems.

You can view the device and single sign-on (SSO) state by running `dsregcmd /status`. The goal is for the device state to show as `AzureAdJoined : YES` and for the SSO state to show `AzureAdPrt : YES`.

RDP sign-in via Azure AD accounts is captured in Event Viewer under the *AAD\Operational* event logs.

### Azure role not assigned

You might get the following error message when you initiate a remote desktop connection to your VM: "Your account is configured to prevent you from using this device. For more info, contact your system administrator."



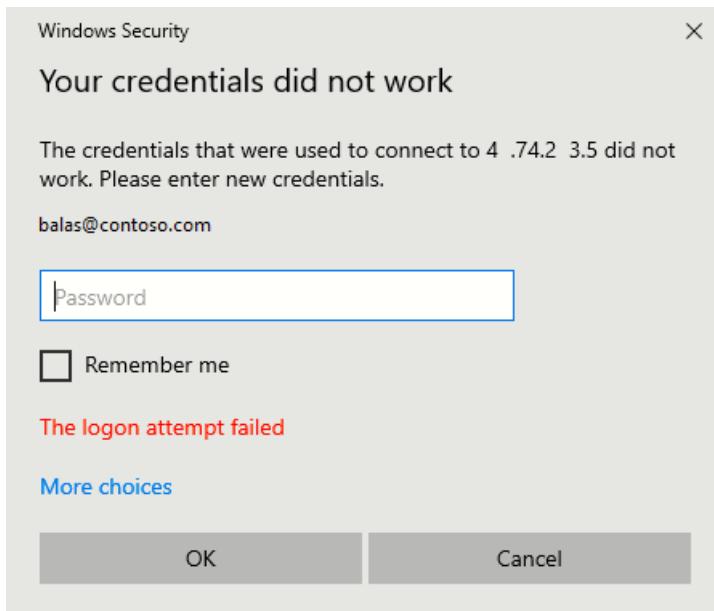
Verify that you've [configured Azure RBAC policies](#) for the VM that grant the user the Virtual Machine Administrator Login or Virtual Machine User Login role.

#### NOTE

If you're having problems with Azure role assignments, see [Troubleshoot Azure RBAC](#).

### Unauthorized client or password change required

You might get the following error message when you initiate a remote desktop connection to your VM: "Your credentials did not work."



Try these solutions:

- The Windows 10 or later PC that you're using to initiate the remote desktop connection must be Azure AD joined, or hybrid Azure AD joined to the same Azure AD directory. For more information about device identity, see the article [What is a device identity?](#).

**NOTE**

Windows 10 Build 20H1 added support for an Azure AD-registered PC to initiate an RDP connection to your VM.

When you're using a PC that's Azure AD registered (not Azure AD joined or hybrid Azure AD joined) as the RDP client to initiate connections to your VM, you must enter credentials in the format

AzureAD\UPN

(for example,

AzureAD\john@contoso.com).

Verify that the AADLoginForWindows extension wasn't uninstalled after the Azure AD join finished.

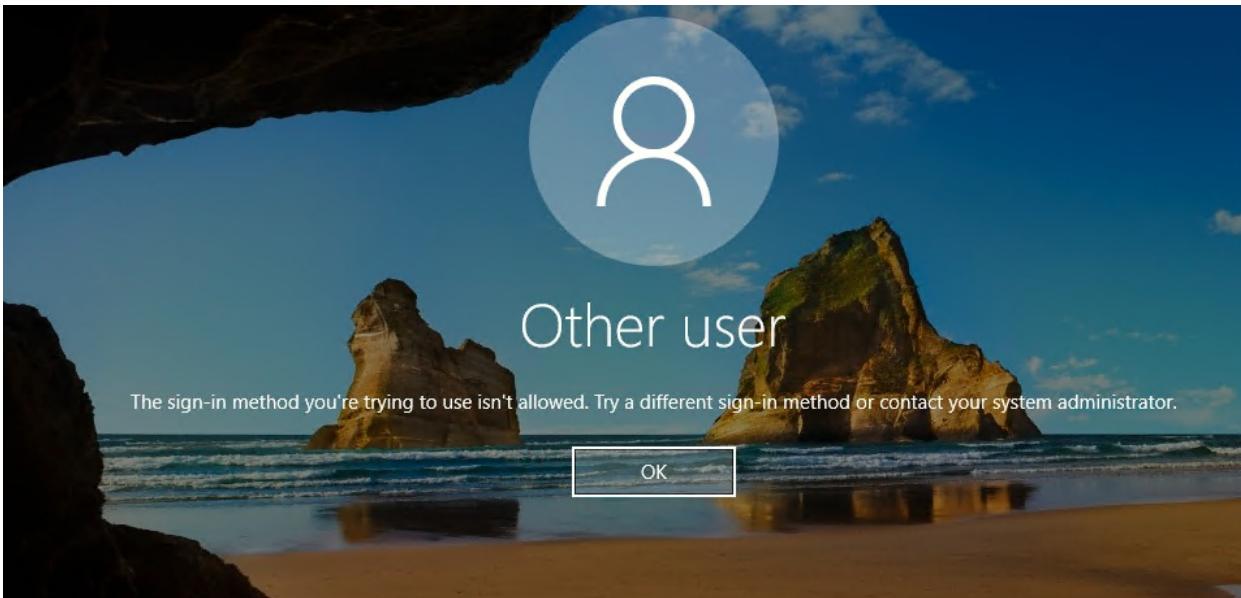
Also, make sure that the security policy **Network security: Allow PKU2U authentication requests to this computer to use online identities** is enabled on both the server *and* the client.

- Verify that the user doesn't have a temporary password. Temporary passwords can't be used to log in to a remote desktop connection.

Sign in with the user account in a web browser. For instance, open the [Azure portal](#) in a private browsing window. If you're prompted to change the password, set a new password. Then try connecting again.

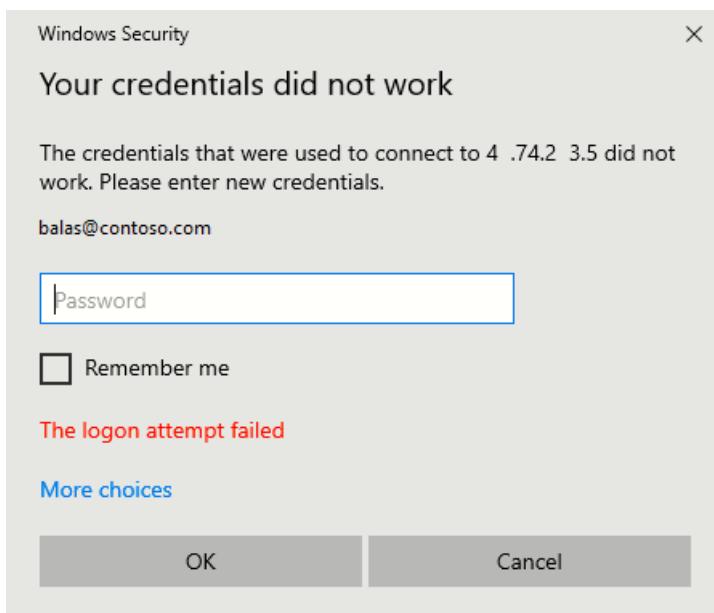
**MFA sign-in method required**

You might see the following error message when you initiate a remote desktop connection to your VM: "The sign-in method you're trying to use isn't allowed. Try a different sign-in method or contact your system administrator."



If you've configured a Conditional Access policy that requires MFA or legacy per-user Enabled/Enforced Azure AD MFA before you can access the resource, you need to ensure that the Windows 10 or later PC that's initiating the remote desktop connection to your VM signs in by using a strong authentication method such as Windows Hello. If you don't use a strong authentication method for your remote desktop connection, you'll see the error.

Another MFA-related error message is the one described previously: "Your credentials did not work."



If you've configured a legacy per-user Enabled/Enforced Azure AD Multi-Factor Authentication setting and you see the error above, you can resolve the problem by removing the per-user MFA setting through these commands:

```
# Get StrongAuthenticationRequirements configure on a user
(Get-MsolUser -UserPrincipalName username@contoso.com).StrongAuthenticationRequirements

# Clear StrongAuthenticationRequirements from a user
$mfa = @()
Set-MsolUser -UserPrincipalName username@contoso.com -StrongAuthenticationRequirements $mfa

# Verify StrongAuthenticationRequirements are cleared from the user
(Get-MsolUser -UserPrincipalName username@contoso.com).StrongAuthenticationRequirements
```

If you haven't deployed Windows Hello for Business and if that isn't an option for now, you can configure a Conditional Access policy that excludes the Azure Windows VM Sign-In app from the list of cloud apps that

require MFA. To learn more about Windows Hello for Business, see [Windows Hello for Business overview](#).

#### NOTE

Windows Hello for Business PIN authentication with RDP has been supported for several versions of Windows 10. Support for biometric authentication with RDP was added in Windows 10 version 1809. Using Windows Hello for Business authentication during RDP is available only for deployments that use a certificate trust model. It's currently not available for a key trust model.

Share your feedback about this feature or report problems with using it on the [Azure AD feedback forum](#).

#### Missing application

If the Azure Windows VM Sign-In application is missing from Conditional Access, make sure that the application isn't in the tenant:

1. Sign in to the Azure portal.
2. Browse to **Azure Active Directory > Enterprise applications**.
3. Remove the filters to see all applications, and search for VM. If you don't see **Azure Windows VM Sign-In** as a result, the service principal is missing from the tenant.

Another way to verify it is via Graph PowerShell:

1. [Install the Graph PowerShell SDK](#) if you haven't already done so.
2. Run `Connect-MgGraph -Scopes "ServicePrincipalEndpoint.ReadWrite.All"`, followed by `"Application.ReadWrite.All"`.
3. Sign in with a Global Admin account.
4. Consent to the permission prompt.
5. Run `Get-MgServicePrincipal -ConsistencyLevel eventual -Search '"DisplayName:Azure Windows VM Sign-In"'`.
  - If this command results in no output and returns you to the PowerShell prompt, you can create the service principal with the following Graph PowerShell command:

```
New-MgServicePrincipal -AppId 372140e0-b3b7-4226-8ef9-d57986796201
```
  - Successful output will show that the Azure Windows VM Sign-In app and its ID were created.
6. Sign out of Graph PowerShell by using the `Disconnect-MgGraph` command.

## Next steps

For more information about Azure AD, see [What is Azure Active Directory?](#).

# Updates and maintenance overview

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article provides an overview of the various update and maintenance options for Azure virtual machines (VMs).

## Automatic OS image upgrade

Enabling [automatic OS image upgrades](#) on your scale set helps ease update management by safely and automatically upgrading the OS disk for all instances in the scale set.

[Automatic OS upgrade](#) has the following characteristics:

- Once configured, the latest OS image published by image publishers is automatically applied to the scale set without user intervention.
- Upgrades batches of instances in a rolling manner each time a new image is published by the publisher.
- Integrates with application health probes and [Application Health extension](#).
- Works for all VM sizes, and for both Windows and Linux images.
- You can opt out of automatic upgrades at any time (OS Upgrades can be initiated manually as well).
- The OS Disk of a VM is replaced with the new OS Disk created with latest image version. Configured extensions and custom data scripts are run, while persisted data disks are retained.
- [Extension sequencing](#) is supported.
- Automatic OS image upgrade can be enabled on a scale set of any size.

## Automatic VM guest patching

Enabling [automatic VM guest patching](#) for your Azure VMs helps ease update management by safely and automatically patching virtual machines to maintain security compliance.

[Automatic VM guest patching](#) has the following characteristics:

- Patches classified as *Critical* or *Security* are automatically downloaded and applied on the VM.
- Patches are applied during off-peak hours in the VM's time zone.
- Patch orchestration is managed by Azure and patches are applied following [availability-first principles](#).
- Virtual machine health, as determined through platform health signals, is monitored to detect patching failures.
- Works for all VM sizes.

## Automatic extension upgrade

[Automatic Extension Upgrade](#) is available for Azure VMs and Azure Virtual Machine Scale Sets. When Automatic Extension Upgrade is enabled on a VM or scale set, the extension is upgraded automatically whenever the extension publisher releases a new version for that extension.

Automatic Extension Upgrade has the following features:

- Supported for Azure VMs and Azure Virtual Machine Scale Sets.
- Upgrades are applied in an availability-first deployment model.
- For a Virtual Machine Scale Set, no more than 20% of the scale set virtual machines will be upgraded in a

single batch. The minimum batch size is one virtual machine.

- Works for all VM sizes, and for both Windows and Linux extensions.
- You can opt out of automatic upgrades at any time.
- Automatic extension upgrade can be enabled on a Virtual Machine Scale Sets of any size.
- Each supported extension is enrolled individually, and you can choose which extensions to upgrade automatically.
- Supported in all public cloud regions.

## Hotpatch

[Hotpatching](#) is a new way to install updates on new Windows Server Azure Edition virtual machines (VMs) that doesn't require a reboot after installation. Hotpatch for Windows Server Azure Edition VMs, has the following benefits:

- Lower workload impact with less reboots
- Faster deployment of updates as the packages are smaller, install faster, and have easier patch orchestration with Azure Update Manager
- Better protection, as the Hotpatch update packages are scoped to Windows security updates that install faster without rebooting

## Azure update management

You can use [Update Management in Azure Automation](#) to manage operating system updates for your Windows and Linux virtual machines in Azure, in on-premises environments, and in other cloud environments. You can quickly assess the status of available updates on all agent machines and manage the process of installing required updates for servers.

## Update management center (preview)

[Update management center \(preview\)](#) is a new-age unified service in Azure to manage and govern updates (Windows and Linux), both on-premises and other cloud platforms, across hybrid environments from a single dashboard. The new functionality provides native and out-of-the-box experience, granular access controls, flexibility to create schedules or take action now, ability to check updates automatically and much more. The enhanced functionality ensures that the administrators have visibility into the health of all systems in the environment. For more information, see [key benefits](#).

## Maintenance control

Manage platform updates, that don't require a reboot, using [maintenance control](#). Azure frequently updates its infrastructure to improve reliability, performance, security or launch new features. Most updates are transparent to users. Some sensitive workloads, like gaming, media streaming, and financial transactions, can't tolerate even few seconds of a VM freezing or disconnecting for maintenance. Maintenance control gives you the option to wait on platform updates and apply them within a 35-day rolling window.

Maintenance control lets you decide when to apply updates to your isolated VMs and Azure dedicated hosts.

With [maintenance control](#), you can:

- Batch updates into one update package.
- Wait up to 35 days to apply updates for Host machines.
- Automate platform updates by configuring a maintenance schedule or by using [Azure Functions](#).
- Maintenance configurations work across subscriptions and resource groups.

## Scheduled events

Scheduled Events is an Azure Metadata Service that gives your application time to prepare for virtual machine (VM) maintenance. It provides information about upcoming maintenance events (for example, reboot) so that your application can prepare for them and limit disruption. It's available for all Azure Virtual Machines types, including PaaS and IaaS on both Windows and Linux.

For information on Scheduled Events, see [Scheduled Events for Windows VMs](#) and [Scheduled Events for Linux](#)

## Next steps

Review the [Availability and scale](#) documentation for more ways to increase the uptime of your applications and services.

# Azure virtual machine scale set automatic OS image upgrades

9/21/2022 • 15 minutes to read • [Edit Online](#)

Enabling automatic OS image upgrades on your scale set helps ease update management by safely and automatically upgrading the OS disk for all instances in the scale set.

Automatic OS upgrade has the following characteristics:

- Once configured, the latest OS image published by image publishers is automatically applied to the scale set without user intervention.
- Upgrades batches of instances in a rolling manner each time a new image is published by the publisher.
- Integrates with application health probes and [Application Health extension](#).
- Works for all VM sizes, and for both Windows and Linux images including custom images through [Azure Compute Gallery](#).
- You can opt out of automatic upgrades at any time (OS Upgrades can be initiated manually as well).
- The OS Disk of a VM is replaced with the new OS Disk created with latest image version. Configured extensions and custom data scripts are run, while persisted data disks are retained.
- [Extension sequencing](#) is supported.
- Can be enabled on a scale set of any size.

## NOTE

Before enabling automatic OS image upgrades, check [requirements section](#) of this documentation.

## How does automatic OS image upgrade work?

An upgrade works by replacing the OS disk of a VM with a new disk created using the latest image version. Any configured extensions and custom data scripts are run on the OS disk, while data disks are retained. To minimize the application downtime, upgrades take place in batches, with no more than 20% of the scale set upgrading at any time.

You can integrate an Azure Load Balancer application health probe or [Application Health extension](#) to track the health of the application after an upgrade. We recommended incorporating an application heartbeat to validate upgrade success.

### Availability-first Updates

The availability-first model for platform orchestrated updates described below ensures that availability configurations in Azure are respected across multiple availability levels.

#### Across regions:

- An update will move across Azure globally in a phased manner to prevent Azure-wide deployment failures.
- A 'phase' can have one or more regions, and an update moves across phases only if eligible VMs in the previous phase update successfully.
- Geo-paired regions will not be updated concurrently and cannot be in the same regional phase.
- The success of an update is measured by tracking the health of a VM post update.

#### Within a region:

- VMs in different Availability Zones are not updated concurrently with the same update.

#### Within a 'set':

- All VMs in a common scale set are not updated concurrently.
- VMs in a common virtual machine scale set are grouped in batches and updated within Update Domain boundaries as described below.

The platform orchestrated updates process is followed for rolling out supported OS platform image upgrades every month. For custom images through Azure Compute Gallery, an image upgrade is only kicked off for a particular Azure region when the new image is published and [replicated](#) to the region of that scale set.

#### Upgrading VMs in a scale set

The region of a scale set becomes eligible to get image upgrades either through the availability-first process for platform images or replicating new custom image versions for Share Image Gallery. The image upgrade is then applied to an individual scale set in a batched manner as follows:

- Before you begin the upgrade process, the orchestrator will ensure that no more than 20% of instances in the entire scale set are unhealthy (for any reason).
- The upgrade orchestrator identifies the batch of VM instances to upgrade, with any one batch having a maximum of 20% of the total instance count, subject to a minimum batch size of one virtual machine. There is no minimum scale set size requirement and scale sets with 5 or fewer instances will have 1 VM per upgrade batch (minimum batch size).
- The OS disk of every VM in the selected upgrade batch is replaced with a new OS disk created from the latest image. All specified extensions and configurations in the scale set model are applied to the upgraded instance.
- For scale sets with configured application health probes or Application Health extension, the upgrade waits up to 5 minutes for the instance to become healthy, before moving on to upgrade the next batch. If an instance does not recover its health in 5 minutes after an upgrade, then by default the previous OS disk for the instance is restored.
- The upgrade orchestrator also tracks the percentage of instances that become unhealthy post an upgrade. The upgrade will stop if more than 20% of upgraded instances become unhealthy during the upgrade process.
- The above process continues until all instances in the scale set have been upgraded.

The scale set OS upgrade orchestrator checks for the overall scale set health before upgrading every batch. While you're upgrading a batch, there could be other concurrent planned or unplanned maintenance activities that could impact the health of your scale set instances. In such cases if more than 20% of the scale set's instances become unhealthy, then the scale set upgrade stops at the end of current batch.

#### NOTE

Automatic OS upgrade does not upgrade the reference image Sku on the scale set. To change the Sku (such as Ubuntu 16.04-LTS to 18.04-LTS), you must update the [scale set model](#) directly with the desired image Sku. Image publisher and offer can't be changed for an existing scale set.

## Supported OS images

Only certain OS platform images are currently supported. Custom images [are supported](#) if the scale set uses custom images through [Azure Compute Gallery](#).

The following platform SKUs are currently supported (and more are added periodically):

PUBLISHER	OS OFFER	SKU
Canonical	UbuntuServer	18.04-LTS
Canonical	UbuntuServer	18.04-LTS-Gen2
Canonical	0001-com-ubuntu-server-focal	20.04-LTS
Canonical	0001-com-ubuntu-server-focal	20.04-LTS-Gen2
MicrosoftCblMariner	Cbl-Mariner	cbl-mariner-1
MicrosoftCblMariner	Cbl-Mariner	1-Gen2
MicrosoftCblMariner	Cbl-Mariner	cbl-mariner-2
MicrosoftCblMariner	Cbl-Mariner	cbl-mariner-2-Gen2
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter
MicrosoftWindowsServer	WindowsServer	2016-Datacenter
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-gs
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-with-Containers
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-with-containers-gs
MicrosoftWindowsServer	WindowsServer	2019-Datacenter
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-Core
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-Core-with-Containers
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-gs
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-with-Containers
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-with-containers-gs
MicrosoftWindowsServer	WindowsServer	2022-Datacenter
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-smalldisk

PUBLISHER	OS OFFER	SKU
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-smalldisk-g2
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-azure-edition
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-core
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-core-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-Datacenter-g2

## Requirements for configuring automatic OS image upgrade

- The *version* property of the image must be set to *latest*.
- Must use application health probes or [Application Health extension](#) for non-Service Fabric scale sets. For Service Fabric requirements, see [Service Fabric requirement](#).
- Use Compute API version 2018-10-01 or higher.
- Ensure that external resources specified in the scale set model are available and updated. Examples include SAS URI for bootstrapping payload in VM extension properties, payload in storage account, reference to secrets in the model, and more.
- For scale sets using Windows virtual machines, starting with Compute API version 2019-03-01, the property *virtualMachineProfile.osProfile.windowsConfiguration.enableAutomaticUpdates* property must set to *false* in the scale set model definition. The *enableAutomaticUpdates* property enables in-VM patching where "Windows Update" applies operating system patches without replacing the OS disk. With automatic OS image upgrades enabled on your scale set, an extra patching process through Windows Update is not required.

### Service Fabric requirements

If you are using Service Fabric, ensure the following conditions are met:

- Service Fabric [durability level](#) is Silver or Gold. If Service Fabric durability is Bronze, only Stateless-only node types supports automatic OS image upgrades).
- The Service Fabric extension on the scale set model definition must have TypeHandlerVersion 1.1 or above.
- Durability level should be the same at the Service Fabric cluster and Service Fabric extension on the scale set model definition.
- An additional health probe or use of application health extension is not required for Silver or Gold durability. Bronze durability with Stateless-only node types requires an additional health probe.
- The property *virtualMachineProfile.osProfile.windowsConfiguration.enableAutomaticUpdates* property must set to *false* in the scale set model definition. The *enableAutomaticUpdates* property enables in-VM patching using "Windows Update" and is not supported on Service Fabric scale sets.

Ensure that durability settings are not mismatched on the Service Fabric cluster and Service Fabric extension, as a mismatch will result in upgrade errors. Durability levels can be modified per the guidelines outlined on [this page](#).

## Automatic OS image upgrade for custom images

Automatic OS image upgrade is supported for custom images deployed through [Azure Compute Gallery](#). Other custom images are not supported for automatic OS image upgrades.

### Additional requirements for custom images

- The setup and configuration process for automatic OS image upgrade is the same for all scale sets as detailed in the [configuration section](#) of this page.
- Scale sets instances configured for automatic OS image upgrades will be upgraded to the latest version of the Azure Compute Gallery image when a new version of the image is published and [replicated](#) to the region of that scale set. If the new image is not replicated to the region where the scale is deployed, the scale set instances will not be upgraded to the latest version. Regional image replication allows you to control the rollout of the new image for your scale sets.
- The new image version should not be excluded from the latest version for that gallery image. Image versions excluded from the gallery image's latest version are not rolled out to the scale set through automatic OS image upgrade.

#### **NOTE**

It can take up to 3 hours for a scale set to trigger the first image upgrade rollout after the scale set is first configured for automatic OS upgrades. This is a one-time delay per scale set. Subsequent image rollouts are triggered on the scale set within 30-60 minutes.

## Configure automatic OS image upgrade

To configure automatic OS image upgrade, ensure that the `automaticOSUpgradePolicy.enableAutomaticOSUpgrade` property is set to `true` in the scale set model definition.

#### **NOTE**

**Upgrade Policy mode** and **Automatic OS Upgrade Policy** are separate settings and control different aspects of the scale set. When there are changes in the scale set template, the Upgrade Policy `mode` will determine what happens to existing instances in the scale set. However, Automatic OS Upgrade Policy `enableAutomaticOSUpgrade` is specific to the OS image and tracks changes the image publisher has made and determines what happens when there is an update to the image.

## REST API

The following example describes how to set automatic OS upgrades on a scale set model:

```
PUT or PATCH on
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachineScaleSets/myScaleSet?api-version=2021-03-01`
```

```
{
  "properties": {
    "upgradePolicy": {
      "automaticOSUpgradePolicy": {
        "enableAutomaticOSUpgrade": true
      }
    }
  }
}
```

## Azure PowerShell

Use the [Update-AzVmss](#) cmdlet to configure automatic OS image upgrades for your scale set. The following example configures automatic upgrades for the scale set named `myScaleSet` in the resource group named `myResourceGroup`.

```
Update-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet" -AutomaticOSUpgrade $true
```

## Azure CLI 2.0

Use [az vmss update](#) to configure automatic OS image upgrades for your scale set. Use Azure CLI 2.0.47 or above. The following example configures automatic upgrades for the scale set named *myScaleSet* in the resource group named *myResourceGroup*:

```
az vmss update --name myScaleSet --resource-group myResourceGroup --set  
UpgradePolicy.AutomaticOSUpgradePolicy.EnableAutomaticOSUpgrade=true
```

### NOTE

After configuring automatic OS image upgrades for your scale set, you must also bring the scale set VMs to the latest scale set model if your scale set uses the 'Manual' [upgrade policy](#).

## Using Application Health Probes

During an OS Upgrade, VM instances in a scale set are upgraded one batch at a time. The upgrade should continue only if the customer application is healthy on the upgraded VM instances. We recommend that the application provides health signals to the scale set OS Upgrade engine. By default, during OS Upgrades the platform considers VM power state and extension provisioning state to determine if a VM instance is healthy after an upgrade. During the OS Upgrade of a VM instance, the OS disk on a VM instance is replaced with a new disk based on latest image version. After the OS Upgrade has completed, the configured extensions are run on these VMs. The application is considered healthy only when all the extensions on the instance are successfully provisioned.

A scale set can optionally be configured with Application Health Probes to provide the platform with accurate information on the ongoing state of the application. Application Health Probes are Custom Load Balancer Probes that are used as a health signal. The application running on a scale set VM instance can respond to external HTTP or TCP requests indicating whether it's healthy. For more information on how Custom Load Balancer Probes work, see to [Understand load balancer probes](#). Application Health Probes are not supported for Service Fabric scale sets. Non-Service Fabric scale sets require either Load Balancer application health probes or [Application Health extension](#).

If the scale set is configured to use multiple placement groups, probes using a [Standard Load Balancer](#) need to be used.

### Configuring a Custom Load Balancer Probe as Application Health Probe on a scale set

As a best practice, create a load balancer probe explicitly for scale set health. The same endpoint for an existing HTTP probe or TCP probe can be used, but a health probe could require different behavior from a traditional load-balancer probe. For example, a traditional load balancer probe could return unhealthy if the load on the instance is too high, but that would not be appropriate for determining the instance health during an automatic OS upgrade. Configure the probe to have a high probing rate of less than two minutes.

The load-balancer probe can be referenced in the *networkProfile* of the scale set and can be associated with either an internal or public facing load-balancer as follows:

```
"networkProfile": {  
    "healthProbe" : {  
        "id": "[concat(variables('lbId'), '/probes/', variables('sshProbeName'))]"  
    },  
    "networkInterfaceConfigurations":  
    ...  
}
```

#### NOTE

When using Automatic OS Upgrades with Service Fabric, the new OS image is rolled out Update Domain by Update Domain to maintain high availability of the services running in Service Fabric. To utilize Automatic OS Upgrades in Service Fabric your cluster node type must be configured to use the Silver Durability Tier or higher. For Bronze Durability tier, automatic OS image upgrade is only supported for Stateless node types. For more information on the durability characteristics of Service Fabric clusters, please see [this documentation](#).

#### Keep credentials up to date

If your scale set uses any credentials to access external resources, such as a VM extension configured to use a SAS token for storage account, then ensure that the credentials are updated. If any credentials, including certificates and tokens, have expired, the upgrade will fail and the first batch of VMs will be left in a failed state.

The recommended steps to recover VMs and re-enable automatic OS upgrade if there's a resource authentication failure are:

- Regenerate the token (or any other credentials) passed into your extension(s).
- Ensure that any credential used from inside the VM to talk to external entities is up to date.
- Update extension(s) in the scale set model with any new tokens.
- Deploy the updated scale set, which will update all VM instances including the failed ones.

## Using Application Health extension

The Application Health extension is deployed inside a virtual machine scale set instance and reports on VM health from inside the scale set instance. You can configure the extension to probe on an application endpoint and update the status of the application on that instance. This instance status is checked by Azure to determine whether an instance is eligible for upgrade operations.

As the extension reports health from within a VM, the extension can be used in situations where external probes such as Application Health Probes (that utilize custom Azure Load Balancer [probes](#)) can't be used.

There are multiple ways of deploying the Application Health extension to your scale sets as detailed in the examples in [this article](#).

## Get the history of automatic OS image upgrades

You can check the history of the most recent OS upgrade performed on your scale set with Azure PowerShell, Azure CLI 2.0, or the REST APIs. You can get history for the last five OS upgrade attempts within the past two months.

#### REST API

The following example uses [REST API](#) to check the status for the scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
GET on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachineScaleSets/myScaleSet/osUpgradeHistory?api-version=2021-03-01`
```

The GET call returns properties similar to the following example output:

```
{  
  "value": [  
    {  
      "properties": {  
        "runningStatus": {  
          "code": "RollingForward",  
          "startTime": "2018-07-24T17:46:06.1248429+00:00",  
          "completedTime": "2018-04-21T12:29:25.0511245+00:00"  
        },  
        "progress": {  
          "successfulInstanceCount": 16,  
          "failedInstanceCount": 0,  
          "inProgressInstanceCount": 4,  
          "pendingInstanceCount": 0  
        },  
        "startedBy": "Platform",  
        "targetImageReference": {  
          "publisher": "MicrosoftWindowsServer",  
          "offer": "WindowsServer",  
          "sku": "2016-Datacenter",  
          "version": "2016.127.20180613"  
        },  
        "rollbackInfo": {  
          "successfullyRolledbackInstanceCount": 0,  
          "failedRolledbackInstanceCount": 0  
        }  
      },  
      "type": "Microsoft.Compute/virtualMachineScaleSets/rollingUpgrades",  
      "location": "westeurope"  
    }  
  ]  
}
```

## Azure PowerShell

Use the [Get-AzVmss](#) cmdlet to check OS upgrade history for your scale set. The following example details how you review the OS upgrade status for a scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
Get-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet" -OSUpgradeHistory
```

## Azure CLI 2.0

Use [az vmss get-os-upgrade-history](#) to check the OS upgrade history for your scale set. Use Azure CLI 2.0.47 or above. The following example details how you review the OS upgrade status for a scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
az vmss get-os-upgrade-history --resource-group myResourceGroup --name myScaleSet
```

# How to get the latest version of a platform OS image?

You can get the available image versions for automatic OS upgrade supported SKUs using the below examples:

## REST API

```
GET on  
`/subscriptions/subscription_id/providers/Microsoft.Compute/locations/{location}/publishers/{publisherName}/  
artifacttypes/vmimage/offers/{offer}/skus/{skus}/versions?api-version=2021-03-01`
```

## Azure PowerShell

```
Get-AzVmImage -Location "westus" -PublisherName "Canonical" -Offer "UbuntuServer" -Skus "16.04-LTS"
```

## Azure CLI 2.0

```
az vm image list --location "westus" --publisher "Canonical" --offer "UbuntuServer" --sku "16.04-LTS" --all
```

# Manually trigger OS image upgrades

With automatic OS image upgrade enabled on your scale set, you do not need to manually trigger image updates on your scale set. The OS upgrade orchestrator will automatically apply the latest available image version to your scale set instances without any manual intervention.

For specific cases where you do not want to wait for the orchestrator to apply the latest image, you can trigger an OS image upgrade manually using the below examples.

### NOTE

Manual trigger of OS image upgrades does not provide automatic rollback capabilities. If an instance does not recover its health after an upgrade operation, its previous OS disk can't be restored.

## REST API

Use the [Start OS Upgrade](#) API call to start a rolling upgrade to move all virtual machine scale set instances to the latest available image OS version. Instances that are already running the latest available OS version are not affected. The following example details how you can start a rolling OS upgrade on a scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
POST on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachineSca  
leSets/myScaleSet/osRollingUpgrade?api-version=2021-03-01`
```

## Azure PowerShell

Use the [Start-AzVmssRollingOSUpgrade](#) cmdlet to check OS upgrade history for your scale set. The following example details how you can start a rolling OS upgrade on a scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
Start-AzVmssRollingOSUpgrade -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet"
```

## Azure CLI 2.0

Use [az vmss rolling-upgrade start](#) to check the OS upgrade history for your scale set. Use Azure CLI 2.0.47 or above. The following example details how you can start a rolling OS upgrade on a scale set named *myScaleSet* in the resource group named *myResourceGroup*.

```
az vmss rolling-upgrade start --resource-group "myResourceGroup" --name "myScaleSet" --subscription  
"subscriptionId"
```

## Next steps

[Learn about the Application Health Extension](#)

# Maintenance control for Azure virtual machine scale sets

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Uniform scale sets

Manage [automatic OS image upgrades](#) for your virtual machine scale sets using maintenance control.

Maintenance control lets you decide when to apply updates to OS disks in your virtual machine scale sets through an easier and more predictable experience.

Maintenance configurations work across subscriptions and resource groups.

The entire workflow comes down to these steps:

- Create a maintenance configuration.
- Associate a virtual machine scale set to a maintenance configuration.
- Enable automatic OS upgrades.

## Limitations

- VMs must be in a scale set.
- User must have **Resource Contributor** access.
- Maintenance duration must be 5 hours or longer in the maintenance configuration.
- Maintenance recurrence must be set to 'Day' in the maintenance configuration

## Management options

You can create and manage maintenance configurations using any of the following options:

- [Azure PowerShell](#)
- [Azure CLI](#)
- [Azure portal](#)

## Next steps

[Virtual machine scale set maintenance control by using PowerShell](#)

# Maintenance control for OS image upgrades on Azure virtual machine scale sets using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Uniform scale sets

Maintenance control lets you decide when to apply automatic guest OS image upgrades to your virtual machine scale sets. This topic covers the Azure PowerShell options for Maintenance control. For more information on using Maintenance control, see [Maintenance control for Azure virtual machine scale sets](#).

## Enable the PowerShell module

Make sure `PowerShellGet` is up to date.

```
Install-Module -Name PowerShellGet -Repository PSGallery -Force
```

Install the `Az.Maintenance` PowerShell module.

```
Install-Module -Name Az.Maintenance
```

If you're installing locally, make sure you open your PowerShell prompt as an administrator.

You may also be asked to confirm that you want to install from an *untrusted repository*. Type `y` or select Yes to All to install the module.

## Connect to an Azure account

Connect to your desired Azure account using `Connect-AzAccount` and `Set-AzAccount`.

```
Connect-AzAccount  
Set-AzContext 00a000aa-0a00-0a0a-00aa-a00a000aaa00  
  
$RGName="myMaintenanceRG"  
$MaintenanceConfig="myMaintenanceConfig"  
$location="eastus2"  
$vmss="myMaintenanceVMSS"
```

## Create a maintenance configuration

Create a resource group as a container for your configuration. In this example, a resource group named `myMaintenanceRG` is created in `eastus2`. If you already have a resource group that you want to use, you can skip this part. Just replace the resource group name with your own in the rest of the examples.

```
New-AzResourceGroup `  
-Location $location `  
-Name $RGName
```

Use `New-AzMaintenanceConfiguration` to create a maintenance configuration. This example creates a maintenance configuration named `myConfig` scoped to the OS image.

```
$config = New-AzMaintenanceConfiguration ` 
    -ResourceGroup $RGName ` 
    -Name $MaintenanceConfig ` 
    -MaintenanceScope OSImage ` 
    -Location $location ` 
    -StartTime "2020-10-01 00:00" ` 
    -TimeZone "Pacific Standard Time" ` 
    -Duration "05:00" ` 
    -RecurEvery "Day"
```

### IMPORTANT

Maintenance **duration** must be *5 hours* or longer. Maintenance **recurrence** must be set to *Day*.

Using `-MaintenanceScope OSImage` ensures that the maintenance configuration is used for controlling updates to the guest OS.

If you try to create a configuration with the same name, but in a different location, you'll get an error. Configuration names must be unique to your resource group.

You can query for available maintenance configurations using [Get-AzMaintenanceConfiguration](#).

```
Get-AzMaintenanceConfiguration | Format-Table -Property Name,Id
```

## Associate your virtual machine scale set to the maintenance configuration

A virtual machine scale set can be associated to any Maintenance configuration regardless of the region and subscription of the Maintenance configuration. By opting in to the Maintenance configuration, new OS image updates for the scale set will be automatically scheduled on the next available maintenance window.

Use [New-AzConfigurationAssignment](#) to associate your virtual machine scale set the maintenance configuration.

```
New-AzConfigurationAssignment ` 
    -ResourceGroupName $RGName ` 
    -Location $location ` 
    -ResourceName $vmss ` 
    -ResourceType VirtualMachineScaleSets ` 
    -ProviderName Microsoft.Compute ` 
    -ConfigurationAssignmentName $config.Name ` 
    -MaintenanceConfigurationId $config.Id
```

## Enable automatic OS upgrade

You can enable automatic OS upgrades for each virtual machine scale set that is going to use maintenance control. For more information about enabling automatic OS upgrades on your virtual machine scale set, see [Azure virtual machine scale set automatic OS image upgrades](#).

## Next steps

[Learn about Maintenance and updates for virtual machines running in Azure](#)

# Maintenance control for OS image upgrades on Azure virtual machine scale sets using Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Maintenance control lets you decide when to apply automatic guest OS image upgrades to your virtual machine scale sets. This topic covers the Azure CLI options for Maintenance control. For more information on using Maintenance control, see [Maintenance control for Azure virtual machine scale sets](#).

## Create a maintenance configuration

Use `az maintenance configuration create` to create a maintenance configuration. This example creates a maintenance configuration named *myConfig* scoped to the osimage.

```
az group create \
--location eastus \
--name myMaintenanceRG
az maintenance configuration create \
-g myMaintenanceRG \
--resource-name myConfig \
--maintenance-scope osimage \
--location eastus
```

Copy the configuration ID from the output to use later.

Using `--maintenance-scope osimage` ensures that the maintenance configuration is used for controlling updates to the guest OS.

If you try to create a configuration with the same name, but in a different location, you will get an error. Configuration names must be unique to your resource group.

You can query for available maintenance configurations using `az maintenance configuration list`.

```
az maintenance configuration list --query "[].{Name:name, ID:id}" -o table
```

### Create a maintenance configuration with a scheduled window

You can also declare a scheduled window when Azure will apply the updates on your resources. This example creates a maintenance configuration named *myConfig* with a scheduled window of 5 hours on the fourth Monday of every month. Once you create a scheduled window, you no longer have to apply the updates manually.

#### IMPORTANT

Maintenance **duration** must be *5 hours* or longer. Maintenance **recurrence** must be set to *Day*.

```
az maintenance configuration create \
-g myMaintenanceRG \
--resource-name myConfig \
--maintenance-scope osimage \
--location eastus \
--maintenance-window-duration "05:00" \
--maintenance-window-recur-every "Day" \
--maintenance-window-start-date-time "2020-12-30 08:00" \
--maintenance-window-time-zone "Pacific Standard Time"
```

## Assign the configuration

Use `az maintenance assignment create` to assign the configuration to your virtual machine scale set.

## Enable automatic OS upgrade

You can enable automatic OS upgrades for each virtual machine scale set that is going to use maintenance control. For more information about enabling automatic OS upgrades on your virtual machine scale set, see [Azure virtual machine scale set automatic OS image upgrades](#).

## Next steps

[Learn about Maintenance and updates for virtual machines running in Azure](#)

# Maintenance control for OS image upgrades on Azure virtual machine scale sets using Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Maintenance control lets you decide when to apply automatic guest OS image upgrades to your virtual machine scale sets. This topic covers the Azure portal options for Maintenance control. For more information on using Maintenance control, see [Maintenance control for Azure virtual machine scale sets](#).

## Create a maintenance configuration

1. Sign in to the Azure portal.
2. Search for **Maintenance Configurations**.



The screenshot shows the Azure search bar with the text "maintenance" typed into it. Below the search bar, the "Services" tab is selected, and the "Maintenance Configurations" item is highlighted with a blue background and white text. The "Services" tab has a blue underline, while other tabs like "Compute" and "Storage" are in grey.

3. Select **Add**.

## Maintenance Configurations

Microsoft (microsoft.onmicrosoft.com)

 Add  Manage view  Refresh  Export to CSV

4. In the Basics tab, choose a subscription and resource group, provide a name for the configuration, choose a region, and select *OS image upgrade* for the scope. Select **Next**.

## Create a maintenance configuration

Basics Schedule Assignments Tags Review + create

Maintenance configurations allow you to batch and delay updates to Azure infrastructure like virtual machines. [Learn more](#)

### Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Contoso
Resource group *	myMaintenanceRG
	<a href="#">Create new</a>

### Instance details

Configuration name *	Demo_Maintenance
Region *	(US) East US 2
Scope *	OS image upgrade

5. In the Schedule tab, declare a scheduled window when Azure will apply the updates on your resources. Set a start date, maintenance window, and recurrence. Once you create a scheduled window, you no longer have to apply the updates manually. Select **Next**.

#### IMPORTANT

Maintenance window **duration** must be *5 hours* or longer. Maintenance **recurrence** must be set to repeat at least once a day.

## Create a maintenance configuration

Basics **Schedule** Assignments Tags Review + create

You can specify a schedule to have granular control over when the updates can happen.

Start date *	06/05/2021	12:00 AM
	(UTC-08:00) Pacific Time (US & Canada)	
Maintenance window	5	Hours
Repeats every *	1	Day
End	<input type="checkbox"/>	

6. In the Assignment tab, assign resources now or skip this step and assign resources after the maintenance configuration deployment. Select **Next**.

7. Add tags and values. Select **Next**.

## Create a maintenance configuration

Basics Schedule Assignments **Tags** Review + create

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more about tags ↗](#)

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name ⓘ	Value ⓘ
project	: Infra
	:

8. Review the summary. Select **Create**.

9. After the deployment is complete, select **Go to resource**.

## Assign the configuration

On the details page of the maintenance configuration, select **Assignments** and then select **Assign resource**.

The screenshot shows the Azure portal interface for managing maintenance configurations. On the left, there's a sidebar with navigation links: Overview, Activity log, Access control (IAM), Tags, Settings, Assignments (which is highlighted in grey), Properties, Locks, and Export template. The main content area has a title 'VM\_Maintenance | Assignments' and a subtitle 'Maintenance Configuration'. At the top right of this area are several buttons: 'Search (Ctrl+ /)', 'Assign resource' (which is highlighted with a red box), 'Unassign resource', 'Run maintenance now', and 'Refresh'. Below these buttons, a message says 'Manage resources assigned to this maintenance configuration. You can assign new resources, see remove them from this configuration.' Underneath is a table with three columns: 'Name ↑↓', 'Type', and 'Maintenance status ↑↓'. The table currently displays 'No results.'

Select the virtual machine scale set resources that you want the maintenance configuration assigned to and select **Ok**.

## Next steps

[Learn about Maintenance and updates for virtual machines running in Azure](#)

# Maintenance control for OS image upgrades on Azure virtual machine scale sets using an ARM template

9/21/2022 • 2 minutes to read • [Edit Online](#)

Maintenance control lets you decide when to apply automatic OS image upgrades to your virtual machine scale sets. For more information on using Maintenance control, see [Maintenance control for Azure virtual machine scale sets](#).

This article explains how you can use an Azure Resource Manager (ARM) template to create a maintenance configuration. You will learn how to:

- Create the configuration
- Assign the configuration to a virtual machine

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

## Create the configuration

While creating the configuration, it is important to note that there are different scopes and each will have unique properties in their creation template. Make sure you are using the right one.

For more information about this Maintenance Configuration template, see [maintenanceConfigurations](#).

### Host and OS image

```
{  
    "type": "Microsoft.Maintenance/maintenanceConfigurations",  
    "apiVersion": "2021-09-01-preview",  
    "name": "string",  
    "location": "string",  
    "tags": {  
        "tagName1": "tagValue1",  
        "tagName2": "tagValue2"  
    },  
    "properties": {  
        "extensionProperties": {},  
        "installPatches": {  
            "linuxParameters": {  
                "classificationsToInclude": [ "string" ],  
                "packageNameMasksToExclude": [ "string" ],  
                "packageNameMasksToInclude": [ "string" ]  
            },  
            "rebootSetting": "string",  
            "tasks": {  
                "postTasks": [  
                    {  
                        "parameters": {},  
                        "source": "string",  
                        "taskScope": "string"  
                    }  
                ],  
                "preTasks": [  
                    {  
                        "parameters": {},  
                        "source": "string",  
                        "taskScope": "string"  
                    }  
                ]  
            },  
            "windowsParameters": {  
                "classificationsToInclude": [ "string" ],  
                "excludeKbsRequiringReboot": "bool",  
                "kbNumbersToExclude": [ "string" ],  
                "kbNumbersToInclude": [ "string" ]  
            }  
        },  
        "maintenanceScope": "string",  
        "maintenanceWindow": {  
            "duration": "string",  
            "expirationDateTime": "string",  
            "recurEvery": "string",  
            "startDateTime": "string",  
            "timeZone": "string"  
        },  
        "namespace": "string",  
        "visibility": "string"  
    }  
}
```

## Assign the configuration

Assign the configuration to a virtual machine.

For more information, see [configurationAssignments](#).

```
{  
  "type": "Microsoft.Maintenance/configurationAssignments",  
  "apiVersion": "2021-09-01-preview",  
  "name": "string",  
  "location": "string",  
  "properties": {  
    "maintenanceConfigurationId": "string",  
    "resourceId": "string"  
  }  
}
```

## Next steps

[Learn about maintenance and updates for virtual machines running in Azure](#)

# Automatic VM guest patching for Azure VMs

9/21/2022 • 15 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

Enabling automatic VM guest patching for your Azure VMs helps ease update management by safely and automatically patching virtual machines to maintain security compliance.

Automatic VM guest patching has the following characteristics:

- Patches classified as *Critical* or *Security* are automatically downloaded and applied on the VM.
- Patches are applied during off-peak hours in the VM's time zone.
- Patch orchestration is managed by Azure and patches are applied following [availability-first principles](#).
- Virtual machine health, as determined through platform health signals, is monitored to detect patching failures.
- Works for all VM sizes.

## How does automatic VM guest patching work?

If automatic VM guest patching is enabled on a VM, then the available *Critical* and *Security* patches are downloaded and applied automatically on the VM. This process kicks off automatically every month when new patches are released. Patch assessment and installation are automatic, and the process includes rebooting the VM as required.

The VM is assessed periodically every few days and multiple times within any 30-day period to determine the applicable patches for that VM. The patches can be installed any day on the VM during off-peak hours for the VM. This automatic assessment ensures that any missing patches are discovered at the earliest possible opportunity.

Patches are installed within 30 days of the monthly patch releases, following availability-first orchestration described below. Patches are installed only during off-peak hours for the VM, depending on the time zone of the VM. The VM must be running during the off-peak hours for patches to be automatically installed. If a VM is powered off during a periodic assessment, the VM will be automatically assessed and applicable patches will be installed automatically during the next periodic assessment (usually within a few days) when the VM is powered on.

Definition updates and other patches not classified as *Critical* or *Security* will not be installed through automatic VM guest patching. To install patches with other patch classifications or schedule patch installation within your own custom maintenance window, you can use [Update Management](#).

### Availability-first Updates

The patch installation process is orchestrated globally by Azure for all VMs that have automatic VM guest patching enabled. This orchestration follows availability-first principles across different levels of availability provided by Azure.

For a group of virtual machines undergoing an update, the Azure platform will orchestrate updates:

#### Across regions:

- A monthly update is orchestrated across Azure globally in a phased manner to prevent global deployment failures.
- A phase can have one or more regions, and an update moves to the next phases only if eligible VMs in a

phase update successfully.

- Geo-paired regions are not updated concurrently and can't be in the same regional phase.
- The success of an update is measured by tracking the VM's health post update. VM Health is tracked through platform health indicators for the VM.

#### Within a region:

- VMs in different Availability Zones are not updated concurrently with the same update.
- VMs that are not part of an availability set are batched on a best effort basis to avoid concurrent updates for all VMs in a subscription.

#### Within an availability set:

- All VMs in a common availability set are not updated concurrently.
- VMs in a common availability set are updated within Update Domain boundaries and VMs across multiple Update Domains are not updated concurrently.

The patch installation date for a given VM may vary month-to-month, as a specific VM may be picked up in a different batch between monthly patching cycles.

#### Which patches are installed?

The patches installed depend on the rollout stage for the VM. Every month, a new global rollout is started where all security and critical patches assessed for an individual VM are installed for that VM. The rollout is orchestrated across all Azure regions in batches (described in the availability-first patching section above).

The exact set of patches to be installed vary based on the VM configuration, including OS type, and assessment timing. It is possible for two identical VMs in different regions to get different patches installed if there are more or less patches available when the patch orchestration reaches different regions at different times. Similarly, but less frequently, VMs within the same region but assessed at different times (due to different Availability Zone or Availability Set batches) might get different patches.

As the Automatic VM Guest Patching does not configure the patch source, two similar VMs configured to different patch sources, such as public repository vs private repository, may also see a difference in the exact set of patches installed.

For OS types that release patches on a fixed cadence, VMs configured to the public repository for the OS can expect to receive the same set of patches across the different rollout phases in a month. For example, Windows VMs configured to the public Windows Update repository.

As a new rollout is triggered every month, a VM will receive at least one patch rollout every month if the VM is powered on during off-peak hours. This process ensures that the VM is patched with the latest available security and critical patches on a monthly basis. To ensure consistency in the set of patches installed, you can configure your VMs to assess and download patches from your own private repositories.

## Supported OS images

#### IMPORTANT

Automatic VM guest patching, on-demand patch assessment and on-demand patch installation are supported only on VMs created from images with the exact combination of publisher, offer and sku from the below supported OS images list. Custom images or any other publisher, offer, sku combinations are not supported. More images are added periodically.

PUBLISHER	OS OFFER	SKU
Canonical	UbuntuServer	16.04-LTS
Canonical	UbuntuServer	18.04-LTS
Canonical	UbuntuServer	18.04-LTS-Gen2
Canonical	0001-com-ubuntu-pro-bionic	pro-18_04-lts
Canonical	0001-com-ubuntu-server-focal	20_04-lts
Canonical	0001-com-ubuntu-server-focal	20_04-lts-gen2
Canonical	0001-com-ubuntu-pro-focal	pro-20_04-lts
microsoftcblmariner	cbl-mariner	cbl-mariner-1
microsoftcblmariner	cbl-mariner	1-gen2
microsoftcblmariner	cbl-mariner	cbl-mariner-2
microsoftcblmariner	cbl-mariner	cbl-mariner-2-gen2
microsoft-aks	aks	aks-engine-ubuntu-1804-202112
Redhat	RHEL	7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7_9, 7-RAW, 7-LVM
Redhat	RHEL	8, 8.1, 8.2, 8_3, 8_4, 8_5, 8-LVM
Redhat	RHEL-RAW	8-raw
OpenLogic	CentOS	7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7_8, 7_9, 7_9-gen2
OpenLogic	centos-lvm	7-lvm
OpenLogic	CentOS	8.0, 8_1, 8_2, 8_3, 8_4, 8_5
OpenLogic	centos-lvm	8-lvm
SUSE	sles-12-sp5	gen1, gen2
SUSE	sles-15-sp2	gen1, gen2
MicrosoftWindowsServer	WindowsServer	2008-R2-SP1
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter
MicrosoftWindowsServer	WindowsServer	2016-Datacenter

PUBLISHER	OS OFFER	SKU
MicrosoftWindowsServer	WindowsServer	2016-datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-Server-Core
MicrosoftWindowsServer	WindowsServer	2016-datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2016-datacenter-with-containers
MicrosoftWindowsServer	WindowsServer	2019-Datacenter
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-Core
MicrosoftWindowsServer	WindowsServer	2019-datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2019-datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2019-datacenter-smalldisk-g2
MicrosoftWindowsServer	WindowsServer	2019-datacenter-with-containers
MicrosoftWindowsServer	WindowsServer	2022-datacenter
MicrosoftWindowsServer	WindowsServer	2022-datacenter-g2
MicrosoftWindowsServer	WindowsServer	2022-datacenter-core
MicrosoftWindowsServer	WindowsServer	2022-datacenter-core-g2
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-smalldisk-g2

## Patch orchestration modes

VMs on Azure now support the following patch orchestration modes:

### **AutomaticByPlatform (Azure-orchestrated patching):**

- This mode is supported for both Linux and Windows VMs.
- This mode enables automatic VM guest patching for the virtual machine and subsequent patch installation is orchestrated by Azure.
- This mode is required for availability-first patching.
- This mode is only supported for VMs that are created using the supported OS platform images above.

- For Windows VMs, setting this mode also disables the native Automatic Updates on the Windows virtual machine to avoid duplication.
- To use this mode on Linux VMs, set the property `osProfile.linuxConfiguration.patchSettings.patchMode=AutomaticByPlatform` in the VM template.
- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByPlatform` in the VM template.

#### AutomaticByOS:

- This mode is supported only for Windows VMs.
- This mode enables Automatic Updates on the Windows virtual machine, and patches are installed on the VM through Automatic Updates.
- This mode does not support availability-first patching.
- This mode is set by default if no other patch mode is specified for a Windows VM.
- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.enableAutomaticUpdates=true`, and set the property `osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByOS` in the VM template.

#### Manual:

- This mode is supported only for Windows VMs.
- This mode disables Automatic Updates on the Windows virtual machine. When deploying a VM using CLI or PowerShell, setting `--enable-auto-updates` to `false` will also set `patchMode` to `manual` and will disable Automatic Updates.
- This mode does not support availability-first patching.
- This mode should be set when using custom patching solutions.
- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.enableAutomaticUpdates=false`, and set the property `osProfile.windowsConfiguration.patchSettings.patchMode=Manual` in the VM template.

#### ImageDefault:

- This mode is supported only for Linux VMs.
- This mode does not support availability-first patching.
- This mode honors the default patching configuration in the image used to create the VM.
- This mode is set by default if no other patch mode is specified for a Linux VM.
- To use this mode on Linux VMs, set the property `osProfile.linuxConfiguration.patchSettings.patchMode=ImageDefault` in the VM template.

#### NOTE

For Windows VMs, the property `osProfile.windowsConfiguration.enableAutomaticUpdates` can only be set when the VM is first created. This impacts certain patch mode transitions. Switching between AutomaticByPlatform and Manual modes is supported on VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=false`. Similarly switching between AutomaticByPlatform and AutomaticByOS modes is supported on VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=true`. Switching between AutomaticByOS and Manual modes is not supported.

## Requirements for enabling automatic VM guest patching

- The virtual machine must have the Azure VM Agent for [Windows](#) or [Linux](#) installed.
- For Linux VMs, the Azure Linux agent must be version 2.2.53.1 or higher. [Update the Linux agent](#) if the

current version is lower than the required version.

- For Windows VMs, the Windows Update service must be running on the virtual machine.
- The virtual machine must be able to access the configured update endpoints. If your virtual machine is configured to use private repositories for Linux or Windows Server Update Services (WSUS) for Windows VMs, the relevant update endpoints must be accessible.
- Use Compute API version 2021-03-01 or higher to access all functionality including on-demand assessment and on-demand patching.
- Custom images are not currently supported.

## Enable automatic VM guest patching

Automatic VM guest patching can be enabled on any Windows or Linux VM that is created from a supported platform image. To enable automatic VM guest patching on a Windows VM, ensure that the property `osProfile.windowsConfiguration.enableAutomaticUpdates` is set to `true` in the VM template definition. This property can only be set when creating the VM. This additional property is not applicable for Linux VMs.

### REST API for Linux VMs

The following example describes how to enable automatic VM guest patching:

```
PUT on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/m  
yVirtualMachine?api-version=2020-12-01`
```

```
{  
    "location": "<location>",  
    "properties": {  
        "osProfile": {  
            "linuxConfiguration": {  
                "provisionVMAgent": true,  
                "patchSettings": {  
                    "patchMode": "AutomaticByPlatform"  
                }  
            }  
        }  
    }  
}
```

### REST API for Windows VMs

The following example describes how to enable automatic VM guest patching:

```
PUT on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/m  
yVirtualMachine?api-version=2020-12-01`
```

```
{  
  "location": "<location>",  
  "properties": {  
    "osProfile": {  
      "windowsConfiguration": {  
        "provisionVMAgent": true,  
        "enableAutomaticUpdates": true,  
        "patchSettings": {  
          "patchMode": "AutomaticByPlatform"  
        }  
      }  
    }  
  }  
}
```

## Azure PowerShell for Windows VMs

Use the [Set-AzVMOperatingSystem](#) cmdlet to enable automatic VM guest patching when creating or updating a VM.

```
Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -Credential $Credential -  
ProvisionVMAgent -EnableAutoUpdate -PatchMode "AutomaticByPlatform"
```

## Azure CLI for Windows VMs

Use [az vm create](#) to enable automatic VM guest patching when creating a new VM. The following example configures automatic VM guest patching for a VM named *myVM* in the resource group named *myResourceGroup*.

```
az vm create --resource-group myResourceGroup --name myVM --image Win2019Datacenter --enable-agent --enable-  
auto-update --patch-mode AutomaticByPlatform
```

To modify an existing VM, use [az vm update](#)

```
az vm update --resource-group myResourceGroup --name myVM --set  
osProfile.windowsConfiguration.enableAutomaticUpdates=true  
osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByPlatform
```

## Azure portal

When creating a VM using the Azure portal, patch orchestration modes can be set under the **Management** tab for both Linux and Windows.

Auto-shutdown

Enable auto-shutdown

Shutdown time

Time zone

Notification before shutdown

Email \*

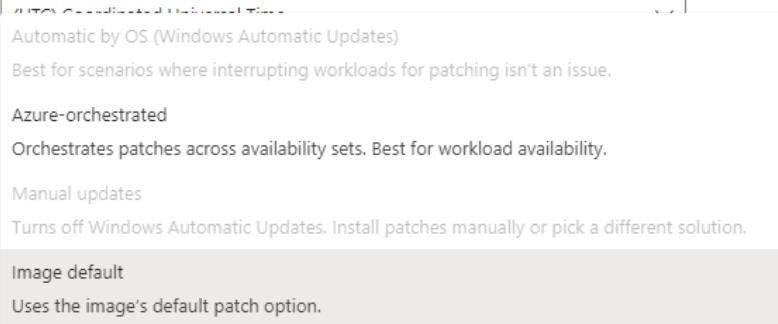
**Backup**

Enable backup

**Guest OS updates**

Patch orchestration options

Some patch orchestration options are not available for this image. [Learn more](#)



[Review + create](#)

[< Previous](#)

[Next : Advanced >](#)

## Enablement and assessment

### NOTE

It can take more than three hours to enable automatic VM guest updates on a VM, as the enablement is completed during the VM's off-peak hours. As assessment and patch installation occur only during off-peak hours, your VM must be also be running during off-peak hours to apply patches.

When automatic VM guest patching is enabled for a VM, a VM extension of type

`Microsoft.CPlat.Core.LinuxPatchExtension` is installed on a Linux VM or a VM extension of type

`Microsoft.CPlat.Core.WindowsPatchExtension` is installed on a Windows VM. This extension does not need to be manually installed or updated, as this extension is managed by the Azure platform as part of the automatic VM guest patching process.

It can take more than three hours to enable automatic VM guest updates on a VM, as the enablement is completed during the VM's off-peak hours. The extension is also installed and updated during off-peak hours for the VM. If the VM's off-peak hours end before enablement can be completed, the enablement process will resume during the next available off-peak time.

Automatic updates are disabled in most scenarios, and patch installation is done through the extension going forward. The following conditions apply.

- If a Windows VM previously had Automatic Windows Update turned on through the AutomaticByOS patch mode, then Automatic Windows Update is turned off for the VM when the extension is installed.
- For Ubuntu VMs, the default automatic updates are disabled automatically when Automatic VM Guest Patching completes enablement.
- For RHEL, automatic updates need to be manually disabled. Execute:

```
systemctl stop packagekit
```

```
systemctl mask packagekit
```

To verify whether automatic VM guest patching has completed and the patching extension is installed on the

VM, you can review the VM's instance view. If the enablement process is complete, the extension will be installed and the assessment results for the VM will be available under `patchStatus`. The VM's instance view can be accessed through multiple ways as described below.

## REST API

```
GET on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/m  
yVirtualMachine/instanceView?api-version=2020-12-01`
```

## Azure PowerShell

Use the [Get-AzVM](#) cmdlet with the `-Status` parameter to access the instance view for your VM.

```
Get-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM" -Status
```

PowerShell currently only provides information on the patch extension. Information about `patchStatus` will also be available soon through PowerShell.

## Azure CLI

Use [az vm get-instance-view](#) to access the instance view for your VM.

```
az vm get-instance-view --resource-group myResourceGroup --name myVM
```

## Understanding the patch status for your VM

The `patchStatus` section of the instance view response provides details on the latest assessment and the last patch installation for your VM.

The assessment results for your VM can be reviewed under the `availablePatchSummary` section. An assessment is periodically conducted for a VM that has automatic VM guest patching enabled. The count of available patches after an assessment is provided under `criticalAndSecurityPatchCount` and `otherPatchCount` results. Automatic VM guest patching will install all patches assessed under the *Critical* and *Security* patch classifications. Any other assessed patch is skipped.

The patch installation results for your VM can be reviewed under the `lastPatchInstallationSummary` section. This section provides details on the last patch installation attempt on the VM, including the number of patches that were installed, pending, failed or skipped. Patches are installed only during the off-peak hours maintenance window for the VM. Pending and failed patches are automatically retried during the next off-peak hours maintenance window.

## Disable automatic VM guest patching

Automatic VM guest patching can be disabled by changing the [patch orchestration mode](#) for the VM.

To disable automatic VM guest patching on a Linux VM, change the patch mode to `ImageDefault`.

To enable automatic VM guest patching on a Windows VM, the property `osProfile.windowsConfiguration.enableAutomaticUpdates` determines which patch modes can be set on the VM and this property can only be set when the VM is first created. This impacts certain patch mode transitions:

- For VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=false`, disable automatic VM guest patching by changing the patch mode to `Manual`.
- For VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=true`, disable automatic VM guest patching by changing the patch mode to `AutomaticByOS`.

- Switching between AutomaticByOS and Manual modes is not supported.

Use the examples from the [enablement](#) section above in this article for API, PowerShell and CLI usage examples to set the required patch mode.

## On-demand patch assessment

If automatic VM guest patching is already enabled for your VM, a periodic patch assessment is performed on the VM during the VM's off-peak hours. This process is automatic and the results of the latest assessment can be reviewed through the VM's instance view as described earlier in this document. You can also trigger an on-demand patch assessment for your VM at any time. Patch assessment can take a few minutes to complete and the status of the latest assessment is updated on the VM's instance view.

### NOTE

On-demand patch assessment does not automatically trigger patch installation. If you have enabled automatic VM guest patching then the assessed and applicable patches for the VM will be installed during the VM's off-peak hours, following the availability-first patching process described earlier in this document.

### REST API

Use the [Assess Patches](#) API to assess available patches for your virtual machine.

```
POST on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/m  
yVirtualMachine/assessPatches?api-version=2020-12-01`
```

### Azure PowerShell

Use the [Invoke-AzVmPatchAssessment](#) cmdlet to assess available patches for your virtual machine.

```
Invoke-AzVmPatchAssessment -ResourceGroupName "myResourceGroup" -VMName "myVM"
```

### Azure CLI

Use [az vm assess-patches](#) to assess available patches for your virtual machine.

```
az vm assess-patches --resource-group myResourceGroup --name myVM
```

## On-demand patch installation

If automatic VM guest patching is already enabled for your VM, a periodic patch installation of Security and Critical patches is performed on the VM during the VM's off-peak hours. This process is automatic and the results of the latest installation can be reviewed through the VM's instance view as described earlier in this document.

You can also trigger an on-demand patch installation for your VM at any time. Patch installation can take a few minutes to complete and the status of the latest installation is updated on the VM's instance view.

You can use on-demand patch installation to install all patches of one or more patch classifications. You can also choose to include or exclude specific packages for Linux or specific KB IDs for Windows. When triggering an on-demand patch installation, ensure that you specify at least one patch classification or at least one patch (package for Linux, KB ID for Windows) in the inclusion list.

### REST API

Use the [Install Patches](#) API to install patches on your virtual machine.

```
POST on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/mVirtualMachine/installPatches?api-version=2020-12-01`
```

Example request body for Linux:

```
{  
    "maximumDuration": "PT1H",  
    "rebootSetting": "IfRequired",  
    "linuxParameters": {  
        "classificationsToInclude": [  
            "Critical",  
            "Security"  
        ]  
    }  
}
```

Example request body for Windows:

```
{  
    "maximumDuration": "PT1H",  
    "rebootSetting": "IfRequired",  
    "windowsParameters": {  
        "classificationsToInclude": [  
            "Critical",  
            "Security"  
        ]  
    }  
}
```

## Azure PowerShell

Use the [Invoke-AzVmInstallPatch](#) cmdlet to install patches on your virtual machine.

Example to install certain packages on a Linux VM:

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -MaximumDuration "PT90M" -  
RebootSetting "Always" -Linux -ClassificationToIncludeForLinux "Security" -PackageNameMaskToInclude  
["package123"] -PackageNameMaskToExclude ["package567"]
```

Example to install all Critical patches on a Windows VM:

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -MaximumDuration "PT2H" -  
RebootSetting "Never" -Windows -ClassificationToIncludeForWindows Critical
```

Example to install all Security patches on a Windows VM, while including and excluding patches with specific KB IDs and excluding any patch that requires a reboot:

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -MaximumDuration "PT90M" -  
RebootSetting "Always" -Windows -ClassificationToIncludeForWindows "Security" -KBNumberToInclude  
["KB1234567", "KB123567"] -KBNumberToExclude ["KB1234702", "KB1234802"] -ExcludeKBsRequiringReboot
```

## Azure CLI

Use [az vm install-patches](#) to install patches on your virtual machine.

Example to install all Critical patches on a Linux VM:

```
az vm install-patches --resource-group myResourceGroup --name myVM --maximum-duration PT2H --reboot-setting IfRequired --classifications-to-include-linux Critical
```

Example to install all Critical and Security patches on a Windows VM, while excluding any patch that requires a reboot:

```
az vm install-patches --resource-group myResourceGroup --name myVM --maximum-duration PT2H --reboot-setting IfRequired --classifications-to-include-win Critical Security --exclude-kbs-requiring-reboot true
```

## Next steps

[Learn more about creating and managing Windows virtual machines](#)

# Hotpatch for new virtual machines

9/21/2022 • 8 minutes to read • [Edit Online](#)

## IMPORTANT

Hotpatch is supported on *Windows Server 2022 Datacenter: Azure Edition (Server Core)*.

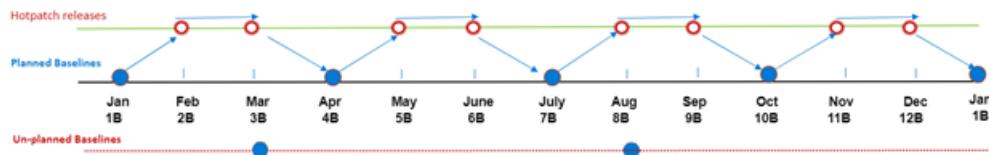
Hotpatching is a new way to install updates on supported *Windows Server Azure Edition* virtual machines (VMs) that doesn't require a reboot after installation. This article covers information about Hotpatch for supported *Windows Server Azure Edition* VMs, which has the following benefits:

- Lower workload impact with less reboots
- Faster deployment of updates as the packages are smaller, install faster, and have easier patch orchestration with Azure Update Manager
- Better protection, as the Hotpatch update packages are scoped to Windows security updates that install faster without rebooting

## How hotpatch works

Hotpatch works by first establishing a baseline with a Windows Update Latest Cumulative Update. Hotpatches are periodically released (for example, on the second Tuesday of the month) that build on that baseline.

Hotpatches will contain updates that don't require a reboot. Periodically (starting at every three months), the baseline is refreshed with a new Latest Cumulative Update.



There are two types of baselines: **Planned baselines** and **unplanned baselines**.

- **Planned baselines** are released on a regular cadence, with hotpatch releases in between. Planned baselines include all the updates in a comparable *Latest Cumulative Update* for that month, and require a reboot.
  - The sample schedule above illustrates four planned baseline releases in a calendar year (five total in the diagram), and eight hotpatch releases.
- **Unplanned baselines** are released when an important update (such as a zero-day fix) is released, and that particular update can't be released as a Hotpatch. When unplanned baselines are released, a hotpatch release will be replaced with an unplanned baseline in that month. Unplanned baselines also include all the updates in a comparable *Latest Cumulative Update* for that month, and also require a reboot.
  - The sample schedule above illustrates two unplanned baselines that would replace the hotpatch releases for those months (the actual number of unplanned baselines in a year isn't known in advance).

## Regional availability

Hotpatch is available in all global Azure regions.

## How to get started

#### NOTE

You can preview onboarding Automanage machine best practices during VM creation in the Azure portal using [this link](#).

To start using Hotpatch on a new VM, follow these steps:

1. Start creating a new VM from the Azure portal

- You can preview onboarding Automanage machine best practices during VM creation in the Azure portal using [this link](#).

2. Supply details during VM creation

- Ensure that a supported *Windows Server Azure Edition* image is selected in the Image dropdown. Use [this guide](#) to determine which images are supported.
- On the Management tab under section 'Guest OS updates', the checkbox for 'Enable hotpatch' will be selected. Patch orchestration options will be set to 'Azure-orchestrated'.
- If you create a VM using [this link](#), on the Management tab under section 'Azure Automanage', select 'Dev/Test' or 'Production' for 'Azure Automanage environment' to evaluate Automanage machine best practices while in preview.

3. Create your new VM

## Patch installation

[Automatic VM Guest Patching](#) is enabled automatically for all VMs created with a supported *Windows Server Azure Edition* image. With automatic VM guest patching enabled:

- Patches classified as Critical or Security are automatically downloaded and applied on the VM.
- Patches are applied during off-peak hours in the VM's time zone.
- Patch orchestration is managed by Azure and patches are applied following [availability-first principles](#).
- Virtual machine health, as determined through platform health signals, is monitored to detect patching failures.

## How does automatic VM guest patching work?

When [Automatic VM Guest Patching](#) is enabled on a VM, the available Critical and Security patches are downloaded and applied automatically. This process kicks off automatically every month when new patches are released. Patch assessment and installation are automatic, and the process includes rebooting the VM as required.

With Hotpatch enabled on supported *Windows Server Azure Edition* VMs, most monthly security updates are delivered as hotpatches that don't require reboots. Latest Cumulative Updates sent on planned or unplanned baseline months will require VM reboots. Additional Critical or Security patches may also be available periodically which may require VM reboots.

The VM is assessed automatically every few days and multiple times within any 30-day period to determine the applicable patches for that VM. This automatic assessment ensures that any missing patches are discovered at the earliest possible opportunity.

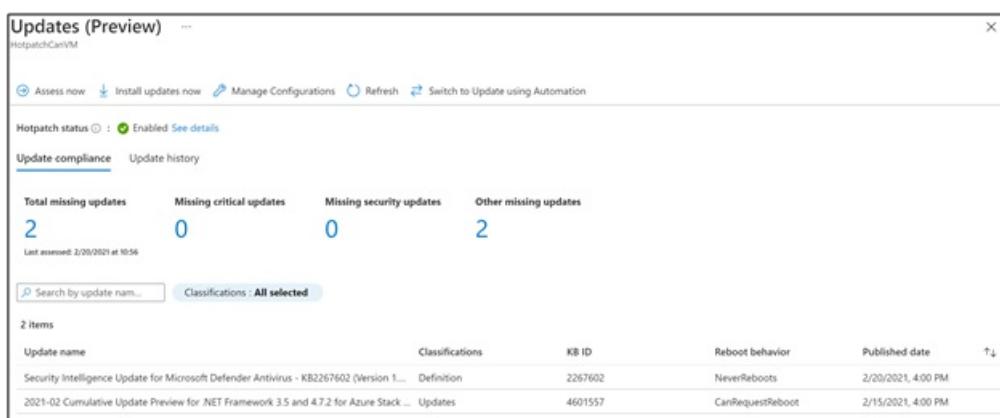
Patches are installed within 30 days of the monthly patch releases, following [availability-first principles](#). Patches are installed only during off-peak hours for the VM, depending on the time zone of the VM. The VM must be running during the off-peak hours for patches to be automatically installed. If a VM is powered off during a periodic assessment, the VM will be assessed and applicable patches will be installed automatically during the next periodic assessment when the VM is powered on. The next periodic assessment usually happens within a few days.

Definition updates and other patches not classified as Critical or Security won't be installed through automatic VM guest patching.

## Understanding the patch status for your VM

To view the patch status for your VM, navigate to the **Guest + host updates** section for your VM in the Azure portal. Under the **Guest OS updates** section, click on 'Go to Hotpatch (Preview)' to view the latest patch status for your VM.

On this screen, you'll see the Hotpatch status for your VM. You can also review if there are any available patches for your VM that haven't been installed. As described in the 'Patch installation' section above, all security and critical updates will be automatically installed on your VM using [Automatic VM Guest Patching](#) and no extra actions are required. Patches with other update classifications aren't automatically installed. Instead, they're viewable in the list of available patches under the 'Update compliance' tab. You can also view the history of update deployments on your VM through the 'Update history'. Update history from the past 30 days is displayed, along with patch installation details.



The screenshot shows the 'Updates (Preview)' window for a VM named 'HotpatchCanVM'. At the top, there are buttons for 'Assess now', 'Install updates now', 'Manage Configurations', 'Refresh', and 'Switch to Update using Automation'. Below this, the 'Hotpatch status' is shown as 'Enabled' with a green checkmark and a 'See details' link. There are two tabs: 'Update compliance' (selected) and 'Update history'. Under 'Update compliance', it shows 'Total missing updates' as 2, with breakdowns: 'Missing critical updates' (0), 'Missing security updates' (0), and 'Other missing updates' (2). The last assessment was on 2/20/2021 at 10:56. A search bar and a classification filter ('All selected') are present. The 'Update history' section shows 2 items:

Update name	Classifications	KB ID	Reboot behavior	Published date
Security Intelligence Update for Microsoft Defender Antivirus - KB2267602 (Version 1...)	Definition	2267602	NeverReboots	2/20/2021, 4:00 PM
2021-02 Cumulative Update Preview for .NET Framework 3.5 and 4.7.2 for Azure Stack ...	Updates	4601557	CanRequestReboot	2/15/2021, 4:00 PM

With automatic VM guest patching, your VM is periodically and automatically assessed for available updates. These periodic assessments ensure that available patches are detected. You can view the results of the assessment on the Updates screen above, including the time of the last assessment. You can also choose to trigger an on-demand patch assessment for your VM at any time using the 'Assess now' option and review the results after assessment completes.

Similar to on-demand assessment, you can also install patches on-demand for your VM using the 'Install updates now' option. Here you can choose to install all updates under specific patch classifications. You can also specify updates to include or exclude by providing a list of individual knowledge base articles. Patches installed on-demand aren't installed using availability-first principles and may require more reboots and VM downtime for update installation.

## Supported updates

Hotpatch covers Windows Security updates and maintains parity with the content of security updates issued to in the regular (non-Hotpatch) Windows update channel.

There are some important considerations to running a supported *Windows Server Azure Edition* VM with Hotpatch enabled. Reboots are still required to install updates that aren't included in the Hotpatch program. Reboots are also required periodically after a new baseline has been installed. These reboots keep the VM in sync with non-security patches included in the latest cumulative update.

- Patches that are currently not included in the Hotpatch program include non-security updates released for Windows, and non-Windows updates (such as .NET patches). These types of patches need to be installed during a baseline month, and will require a reboot.

# Frequently asked questions

## What is hotpatching?

- Hotpatching is a new way to install updates on a supported *Windows Server Azure Edition* VM in Azure that doesn't require a reboot after installation. It works by patching the in-memory code of running processes without the need to restart the process.

## How does hotpatching work?

- Hotpatching works by establishing a baseline with a Windows Update Latest Cumulative Update, then builds upon that baseline with updates that don't require a reboot to take effect. The baseline is updated periodically with a new cumulative update. The cumulative update includes all security and quality updates and requires a reboot.

## Why should I use Hotpatch?

- When you use Hotpatch on a supported *Windows Server Azure Edition* image, your VM will have higher availability (fewer reboots), and faster updates (smaller packages that are installed faster without the need to restart processes). This process results in a VM that is always up to date and secure.

## What types of updates are covered by Hotpatch?

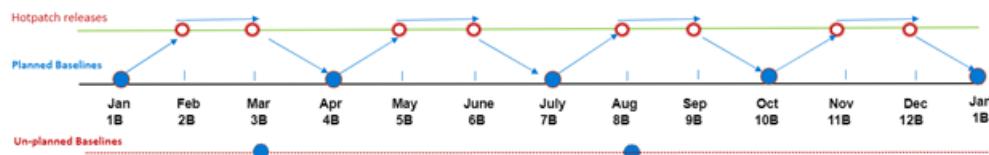
- Hotpatch currently covers Windows security updates.

## When will I receive the first Hotpatch update?

- Hotpatch updates are typically released on the second Tuesday of each month. For more information, see below.

## What will the Hotpatch schedule look like?

- Hotpatching works by establishing a baseline with a Windows Update Latest Cumulative Update, then builds upon that baseline with Hotpatch updates released monthly. Baselines will be released starting out every three months. See the image below for an example of an annual three-month schedule (including example unplanned baselines due to zero-day fixes).



## Are reboots still needed for a VM enrolled in Hotpatch?

- Reboots are still required to install updates not included in the Hotpatch program, and are required periodically after a baseline (Windows Update Latest Cumulative Update) has been installed. This reboot will keep your VM in sync with all the patches included in the cumulative update. Baselines (which require a reboot) will start out on a three-month cadence and increase over time.

## Are my applications affected when a Hotpatch update is installed?

- Because Hotpatch patches the in-memory code of running processes without the need to restart the process, your applications will be unaffected by the patching process. Note that this is separate from any potential performance and functionality implications of the patch itself.

## Can I turn off Hotpatch on my VM?

- You can turn off Hotpatch on a VM via the Azure portal. Turning off Hotpatch will unenroll the VM from Hotpatch, which reverts the VM to typical update behavior for Windows Server. Once you unenroll from Hotpatch on a VM, you can re-enroll that VM when the next Hotpatch baseline is released.

## Can I upgrade from my existing Windows Server OS?

- Yes, upgrading from existing versions of Windows Server (such as Windows Server 2016 or Windows Server

2019) to *Windows Server 2022 Datacenter: Azure Edition* is supported.

### How can I get troubleshooting support for Hotpatching?

- You can file a [technical support case ticket](#). For the Service option, search for and select **Virtual Machine running Windows** under Compute. Select **Azure Features** for the problem type and **Automatic VM Guest Patching** for the problem subtype.

## Next steps

- Learn about [Azure Update Management](#)
- Learn more about [Automatic VM Guest Patching](#)
- Learn more about [Automanage for Windows Server](#)

# Automatic Extension Upgrade for VMs and Scale Sets in Azure

9/21/2022 • 6 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Automatic Extension Upgrade is available for Azure VMs and Azure Virtual Machine Scale Sets. When Automatic Extension Upgrade is enabled on a VM or scale set, the extension is upgraded automatically whenever the extension publisher releases a new version for that extension.

Automatic Extension Upgrade has the following features:

- Supported for Azure VMs and Azure Virtual Machine Scale Sets.
- Upgrades are applied in an availability-first deployment model (detailed below).
- For a Virtual Machine Scale Set, no more than 20% of the scale set virtual machines will be upgraded in a single batch. The minimum batch size is one virtual machine.
- Works for all VM sizes, and for both Windows and Linux extensions.
- You can opt out of automatic upgrades at any time.
- Automatic extension upgrade can be enabled on a Virtual Machine Scale Sets of any size.
- Each supported extension is enrolled individually, and you can choose which extensions to upgrade automatically.
- Supported in all public cloud regions.

## How does Automatic Extension Upgrade work?

The extension upgrade process replaces the existing extension version on a VM with a new version of the same extension when published by the extension publisher. The health of the VM is monitored after the new extension is installed. If the VM is not in a healthy state within 5 minutes of the upgrade completion, the extension version is rolled back to the previous version.

A failed extension update is automatically retried. A retry is attempted every few days automatically without user intervention.

### Availability-first Updates

The availability-first model for platform orchestrated updates ensures that availability configurations in Azure are respected across multiple availability levels.

For a group of virtual machines undergoing an update, the Azure platform will orchestrate updates:

#### Across regions:

- An update will move across Azure globally in a phased manner to prevent Azure-wide deployment failures.
- A 'phase' can have one or more regions, and an update moves across phases only if eligible VMs in the previous phase update successfully.
- Geo-paired regions will not be updated concurrently and cannot be in the same regional phase.
- The success of an update is measured by tracking the health of a VM post update. VM health is tracked through platform health indicators for the VM. For Virtual Machine Scale Sets, the VM health is tracked through application health probes or the Application Health extension, if applied to the scale set.

#### Within a region:

- VMs in different Availability Zones are not updated concurrently with the same update.
- Single VMs that are not part of an availability set are batched on a best effort basis to avoid concurrent updates for all VMs in a subscription.

#### Within a 'set':

- All VMs in a common availability set or scale set are not updated concurrently.
- VMs in a common availability set are updated within Update Domain boundaries and VMs across multiple Update Domains are not updated concurrently.
- VMs in a common virtual machine scale set are grouped in batches and updated within Update Domain boundaries.

#### Upgrade process for Virtual Machine Scale Sets

1. Before beginning the upgrade process, the orchestrator will ensure that no more than 20% of VMs in the entire scale set are unhealthy (for any reason).
2. The upgrade orchestrator identifies the batch of VM instances to upgrade. An upgrade batch can have a maximum of 20% of the total VM count, subject to a minimum batch size of one virtual machine.
3. For scale sets with configured application health probes or Application Health extension, the upgrade waits up to 5 minutes (or the defined health probe configuration) for the VM to become healthy before upgrading the next batch. If a VM does not recover its health after an upgrade, then by default the previous extension version on the VM is reinstalled.
4. The upgrade orchestrator also tracks the percentage of VMs that become unhealthy after an upgrade. The upgrade will stop if more than 20% of upgraded instances become unhealthy during the upgrade process.

The above process continues until all instances in the scale set have been upgraded.

The scale set upgrade orchestrator checks for the overall scale set health before upgrading every batch. While upgrading a batch, there could be other concurrent planned or unplanned maintenance activities that could impact the health of your scale set virtual machines. In such cases, if more than 20% of the scale set's instances become unhealthy, then the scale set upgrade stops at the end of current batch.

## Supported extensions

Automatic Extension Upgrade supports the following extensions (and more are added periodically):

- Dependency Agent – [Linux](#) and [Windows](#)
- [Application Health Extension](#) – Linux and Windows
- [Guest Configuration Extension](#) – Linux and Windows
- Key Vault – [Linux](#) and [Windows](#)
- [Azure Monitor Agent](#)
- [DSC extension for Linux](#)

## Enabling Automatic Extension Upgrade

To enable Automatic Extension Upgrade for an extension, you must ensure the property `enableAutomaticUpgrade` is set to `true` and added to every extension definition individually.

#### REST API for Virtual Machines

To enable automatic extension upgrade for an extension (in this example the Dependency Agent extension) on an Azure VM, use the following:

```
PUT on  
`/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachi  
nes/<vmName>/extensions/<extensionName>?api-version=2019-12-01`
```

```
{  
    "name": "extensionName",  
    "type": "Microsoft.Compute/virtualMachines/extensions",  
    "location": "<location>",  
    "properties": {  
        "autoUpgradeMinorVersion": true,  
        "enableAutomaticUpgrade": true,  
        "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",  
        "type": "DependencyAgentWindows",  
        "typeHandlerVersion": "9.5"  
    }  
}
```

## REST API for Virtual Machine Scale Sets

Use the following to add the extension to the scale set model:

```
PUT on  
`/subscriptions/<subscriptionId>/resourceGroups/<resourceGroupName>/providers/Microsoft.Compute/virtualMachi  
neScaleSets/<vmssName>?api-version=2019-12-01`
```

```
{  
    "location": "<location>",  
    "properties": {  
        "virtualMachineProfile": {  
            "extensionProfile": {  
                "extensions": [  
                    {  
                        "name": "<extensionName>",  
                        "properties": {  
                            "autoUpgradeMinorVersion": true,  
                            "enableAutomaticUpgrade": true,  
                            "publisher": "Microsoft.Azure.Monitoring.DependencyAgent",  
                            "type": "DependencyAgentWindows",  
                            "typeHandlerVersion": "9.5"  
                        }  
                    }  
                ]  
            }  
        }  
    }  
}
```

## Azure PowerShell for Virtual Machines

Use the [Set-AzVMExtension](#) cmdlet:

```
Set-AzVMExtension -ExtensionName "Microsoft.Azure.Monitoring.DependencyAgent" `  
    -ResourceGroupName "myResourceGroup" `  
    -VMName "myVM" `  
    -Publisher "Microsoft.Azure.Monitoring.DependencyAgent" `  
    -ExtensionType "DependencyAgentWindows" `  
    -TypeHandlerVersion 9.5 `  
    -Location WestUS `  
    -EnableAutomaticUpgrade $true
```

## Azure PowerShell for Virtual Machine Scale Sets

Use the [Add-AzVmssExtension](#) cmdlet to add the extension to the scale set model:

```
Add-AzVmssExtension -VirtualMachineScaleSet $vmss  
-Name "Microsoft.Azure.Monitoring.DependencyAgent"  
-Publisher "Microsoft.Azure.Monitoring.DependencyAgent"  
-Type "DependencyAgentWindows"  
-TypeHandlerVersion 9.5  
-EnableAutomaticUpgrade $true
```

Update the scale set using [Update-AzVmss](#) after adding the extension.

## Azure CLI for Virtual Machines

Use the [az vm extension set](#) cmdlet:

```
az vm extension set \  
--resource-group myResourceGroup \  
--vm-name myVM \  
--name DependencyAgentLinux \  
--publisher Microsoft.Azure.Monitoring.DependencyAgent \  
--version 9.5 \  
--enable-auto-upgrade true
```

## Azure CLI for Virtual Machine Scale Sets

Use the [az vmss extension set](#) cmdlet to add the extension to the scale set model:

```
az vmss extension set \  
--resource-group myResourceGroup \  
--vmss-name myVMSS \  
--name DependencyAgentLinux \  
--publisher Microsoft.Azure.Monitoring.DependencyAgent \  
--version 9.5 \  
--enable-auto-upgrade true
```

## Extension upgrades with multiple extensions

A VM or Virtual Machine Scale Set can have multiple extensions with automatic extension upgrade enabled. The same VM or scale set can also have other extensions without automatic extension upgrade enabled.

If multiple extension upgrades are available for a virtual machine, the upgrades may be batched together, but each extension upgrade is applied individually on a virtual machine. A failure on one extension does not impact the other extension(s) that may be upgrading. For example, if two extensions are scheduled for an upgrade, and the first extension upgrade fails, the second extension will still be upgraded.

Automatic Extension Upgrades can also be applied when a VM or virtual machine scale set has multiple extensions configured with [extension sequencing](#). Extension sequencing is applicable for the first-time deployment of the VM, and any future extension upgrades on an extension are applied independently.

## Next steps

[Learn about the Application Health Extension](#)

# Update Management overview

9/21/2022 • 12 minutes to read • [Edit Online](#)

You can use Update Management in Azure Automation to manage operating system updates for your Windows and Linux virtual machines in Azure, physical or VMs in on-premises environments, and in other cloud environments. You can quickly assess the status of available updates and manage the process of installing required updates for your machines reporting to Update Management.

As a service provider, you may have onboarded multiple customer tenants to [Azure Lighthouse](#). Update Management can be used to assess and schedule update deployments to machines in multiple subscriptions in the same Azure Active Directory (Azure AD) tenant, or across tenants using Azure Lighthouse.

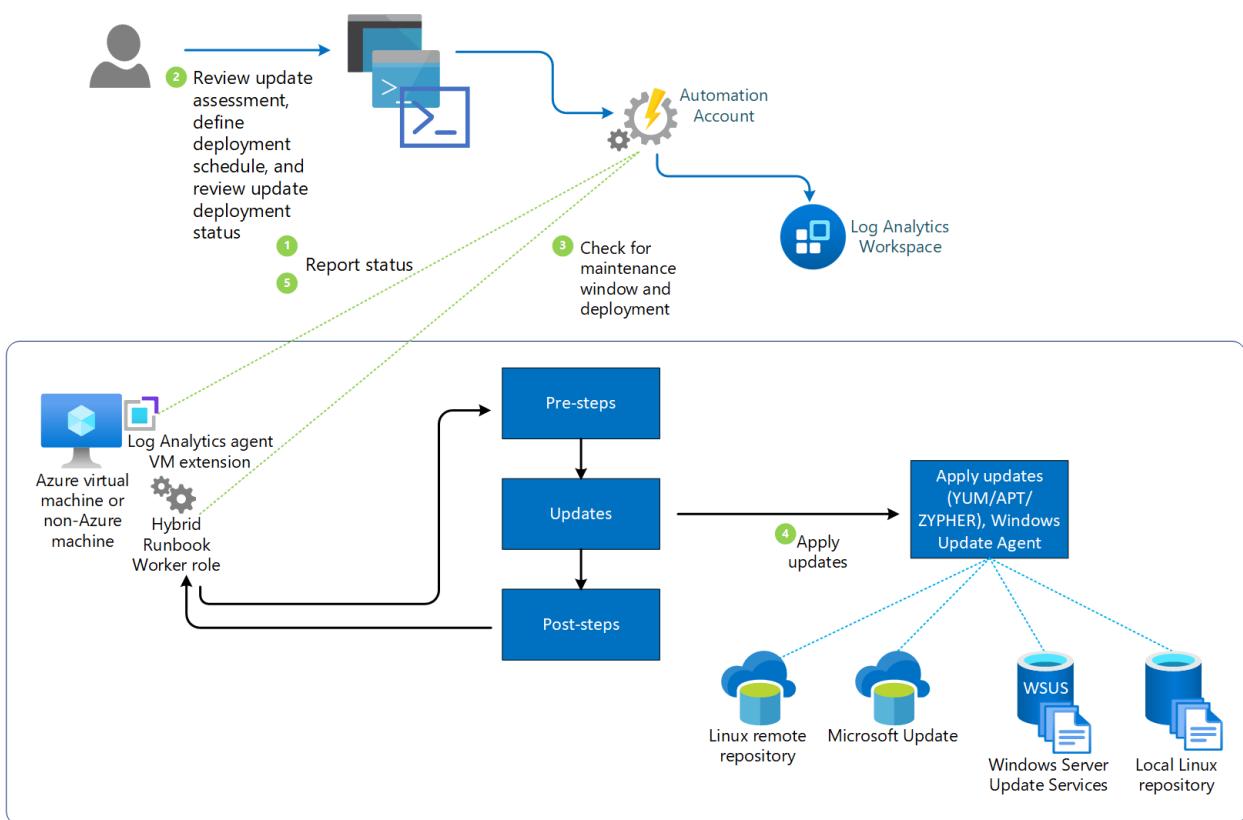
Microsoft offers other capabilities to help you manage updates for your Azure VMs or Azure virtual machine scale sets that you should consider as part of your overall update management strategy.

- If you are interested in automatically assessing and updating your Azure virtual machines to maintain security compliance with *Critical* and *Security* updates released each month, review [Automatic VM guest patching](#). This is an alternative update management solution for your Azure VMs to auto-update them during off-peak hours, including VMs within an availability set, compared to managing update deployments to those VMs from Update Management in Azure Automation.
- If you manage Azure virtual machine scale sets, review how to perform [automatic OS image upgrades](#) to safely and automatically upgrade the OS disk for all instances in the scale set.

Before deploying Update Management and enabling your machines for management, make sure that you understand the information in the following sections.

## About Update Management

The following diagram illustrates how Update Management assesses and applies security updates to all connected Windows Server and Linux servers.



Update Management integrates with Azure Monitor Logs to store update assessments and update deployment results as log data, from assigned Azure and non-Azure machines. To collect this data, the Automation Account and Log Analytics workspace are linked together, and the Log Analytics agent for Windows and Linux is required on the machine and configured to report to this workspace.

Update Management supports collecting information about system updates from agents in a System Center Operations Manager management group connected to the workspace. Having a machine registered for Update Management in more than one Log Analytics workspace (also referred to as multihoming) isn't supported.

The following table summarizes the supported connected sources with Update Management.

CONNECTED SOURCE	SUPPORTED	DESCRIPTION
Windows	Yes	<p>Update Management collects information about system updates from Windows machines with the Log Analytics agent and installation of required updates.</p> <p>Machines need to report to Microsoft Update or Windows Server Update Services (WSUS).</p>
Linux	Yes	<p>Update Management collects information about system updates from Linux machines with the Log Analytics agent and installation of required updates on supported distributions.</p> <p>Machines need to report to a local or remote repository.</p>

CONNECTED SOURCE	SUPPORTED	DESCRIPTION
Operations Manager management group	Yes	<p>Update Management collects information about software updates from agents in a connected management group.</p> <p>A direct connection from the Operations Manager agent to Azure Monitor logs isn't required. Log data is forwarded from the management group to the Log Analytics workspace.</p>

The machines assigned to Update Management report how up to date they are based on what source they are configured to synchronize with. Windows machines need to be configured to report to either [Windows Server Update Services](#) or [Microsoft Update](#), and Linux machines need to be configured to report to a local or public repository. You can also use Update Management with Microsoft Endpoint Configuration Manager, and to learn more see [Integrate Update Management with Windows Endpoint Configuration Manager](#).

If the Windows Update Agent (WUA) on the Windows machine is configured to report to WSUS, depending on when WSUS last synchronized with Microsoft Update, the results might differ from what Microsoft Update shows. This behavior is the same for Linux machines that are configured to report to a local repo instead of a public repo. On a Windows machine, the compliance scan is run every 12 hours by default. For a Linux machine, the compliance scan is performed every hour by default. If the Log Analytics agent is restarted, a compliance scan is started within 15 minutes. When a machine completes a scan for update compliance, the agent forwards the information in bulk to Azure Monitor Logs.

You can deploy and install software updates on machines that require the updates by creating a scheduled deployment. Updates classified as *Optional* aren't included in the deployment scope for Windows machines. Only required updates are included in the deployment scope.

The scheduled deployment defines which target machines receive the applicable updates. It does so either by explicitly specifying certain machines or by selecting a [computer group](#) that's based on log searches of a specific set of machines (or based on an [Azure query](#) that dynamically selects Azure VMs based on specified criteria). These groups differ from [scope configuration](#), which is used to control the targeting of machines that receive the configuration to enable Update Management. This prevents them from performing and reporting update compliance, and install approved required updates.

While defining a deployment, you also specify a schedule to approve and set a time period during which updates can be installed. This period is called the maintenance window. A 10-minute span of the maintenance window is reserved for reboots, assuming one is needed and you selected the appropriate reboot option. If patching takes longer than expected and there's less than 10 minutes in the maintenance window, a reboot won't occur.

After an update package is scheduled for deployment, it takes 2 to 3 hours for the update to show up for Linux machines for assessment. For Windows machines, it takes 12 to 15 hours for the update to show up for assessment after it's been released. Before and after update installation, a scan for update compliance is performed and the log data results is forwarded to the workspace.

Updates are installed by runbooks in Azure Automation. You can't view these runbooks, and they don't require any configuration. When an update deployment is created, it creates a schedule that starts a master update runbook at the specified time for the included machines. The master runbook starts a child runbook on each agent that initiates the installation of the required updates with the Windows Update agent on Windows, or the applicable command on supported Linux distro.

At the date and time specified in the update deployment, the target machines execute the deployment in parallel. Before installation, a scan is run to verify that the updates are still required. For WSUS client machines, if the

updates aren't approved in WSUS, update deployment fails.

## Limits

For limits that apply to Update Management, see [Azure Automation service limits](#).

## Permissions

To create and manage update deployments, you need specific permissions. To learn about these permissions, see [Role-based access - Update Management](#).

## Update Management components

Update Management uses the resources described in this section. These resources are automatically added to your Automation account when you enable Update Management.

### Hybrid Runbook Worker groups

After you enable Update Management, any Windows machine that's directly connected to your Log Analytics workspace is automatically configured as a system Hybrid Runbook Worker to support the runbooks that support Update Management.

Each Windows machine that's managed by Update Management is listed in the Hybrid worker groups pane as a System hybrid worker group for the Automation account. The groups use the `Hostname_FQDN_GUID` naming convention. You can't target these groups with runbooks in your account. If you try, the attempt fails. These groups are intended to support only Update Management. To learn more about viewing the list of Windows machines configured as a Hybrid Runbook Worker, see [view Hybrid Runbook Workers](#).

You can add the Windows machine to a user Hybrid Runbook Worker group in your Automation account to support Automation runbooks if you use the same account for Update Management and the Hybrid Runbook Worker group membership. This functionality was added in version 7.2.12024.0 of the Hybrid Runbook Worker.

### External dependencies

Azure Automation Update Management depends on the following external dependencies to deliver software updates.

- Windows Server Update Services (WSUS) or Microsoft Update is needed for software updates packages and for the software updates applicability scan on Windows-based machines.
- The Windows Update Agent (WUA) client is required on Windows-based machines so that they can connect to the WSUS server or Microsoft Update.
- A local or remote repository to retrieve and installs OS updates on Linux-based machines.

### Management packs

The following management packs are installed on the machines managed by Update Management. If your Operations Manager management group is [connected to a Log Analytics workspace](#), the management packs are installed in the Operations Manager management group. You don't need to configure or manage these management packs.

- Microsoft System Center Advisor Update Assessment Intelligence Pack  
(`Microsoft.IntelligencePacks.UpdateAssessment`)
- `Microsoft.IntelligencePack.UpdateAssessment.Configuration`  
(`Microsoft.IntelligencePack.UpdateAssessment.Configuration`)
- Update Deployment MP

#### **NOTE**

If you have an Operations Manager 1807 or 2019 management group connected to a Log Analytics workspace with agents configured in the management group to collect log data, you need to override the parameter `IsAutoRegistrationEnabled` and set it to `True` in the `Microsoft.IntelligencePacks.AzureAutomation.HybridAgent.Init` rule.

For more information about updates to management packs, see [Connect Operations Manager to Azure Monitor logs](#).

#### **NOTE**

For Update Management to fully manage machines with the Log Analytics agent, you must update to the Log Analytics agent for Windows or the Log Analytics agent for Linux. To learn how to update the agent, see [How to upgrade an Operations Manager agent](#). In environments that use Operations Manager, you must be running System Center Operations Manager 2012 R2 UR 14 or later.

## Data collection frequency

Update Management scans managed machines for data using the following rules. It can take between 30 minutes and 6 hours for the dashboard to display updated data from managed machines.

- Each Windows machine - Update Management does a scan twice per day for each machine.
- Each Linux machine - Update Management does a scan every hour.

The average data usage by Azure Monitor logs for a machine using Update Management is approximately 25 MB per month. This value is only an approximation and is subject to change, depending on your environment. We recommend that you monitor your environment to keep track of your exact usage. For more information about analyzing Azure Monitor Logs data usage, see [Azure Monitor Logs pricing details](#).

## Update classifications

The following table defines the classifications that Update Management supports for Windows updates.

CLASSIFICATION	DESCRIPTION
Critical updates	An update for a specific problem that addresses a critical, non-security-related bug.
Security updates	An update for a product-specific, security-related issue.
Update rollups	A cumulative set of hotfixes that are packaged together for easy deployment.
Feature packs	New product features that are distributed outside a product release.
Service packs	A cumulative set of hotfixes that are applied to an application.
Definition updates	An update to virus or other definition files.
Tools	A utility or feature that helps complete one or more tasks.

Classification	Description
Updates	An update to an application or file that currently is installed.

The next table defines the supported classifications for Linux updates.

Classification	Description
Critical and security updates	Updates for a specific problem or a product-specific, security-related issue.
Other updates	All other updates that aren't critical in nature or that aren't security updates.

#### NOTE

Update classification for Linux machines is only available when used in supported Azure public cloud regions. There is no classification of Linux updates when using Update Management in the following national cloud regions:

- Azure US Government
- 21Vianet in China

Instead of being classified, updates are reported under the **Other updates** category.

Update Management uses data published by the supported distributions, specifically their released [OVAL](#) (Open Vulnerability and Assessment Language) files. Because internet access is restricted from these national clouds, Update Management cannot access the files.

For Linux, Update Management can distinguish between critical updates and security updates in the cloud under classification **Security** and **Others**, while displaying assessment data due to data enrichment in the cloud. For patching, Update Management relies on classification data available on the machine. Unlike other distributions, CentOS does not have this information available in the RTM version. If you have CentOS machines configured to return security data for the following command, Update Management can patch based on classifications.

```
sudo yum -q --security check-update
```

There's currently no supported method to enable native classification-data availability on CentOS. At this time, limited support is provided to customers who might have enabled this feature on their own.

To classify updates on Red Hat Enterprise version 6, you need to install the yum-security plugin. On Red Hat Enterprise Linux 7, the plugin is already a part of yum itself and there's no need to install anything. For more information, see the following Red Hat [knowledge article](#).

When you schedule an update to run on a Linux machine, that for example is configured to install only updates matching the **Security** classification, the updates installed might be different from, or are a subset of, the updates matching this classification. When an assessment of OS updates pending for your Linux machine is performed, [Open Vulnerability and Assessment Language](#) (OVAL) files provided by the Linux distro vendor is used by Update Management for classification.

Categorization is done for Linux updates as **Security** or **Others** based on the OVAL files, which includes updates addressing security issues or vulnerabilities. But when the update schedule is run, it executes on the Linux machine using the appropriate package manager like YUM, APT, or ZYPPER to install them. The package manager for the Linux distro may have a different mechanism to classify updates, where the results may differ from the ones obtained from OVAL files by Update Management. To manually check the machine and understand which updates are security relevant by your package manager, see [Troubleshoot Linux update](#)

deployment.

#### NOTE

During update assessment, the classification of missing updates as Security and Critical may not work correctly for Linux distros supported by Update Management. This is a result of an issue identified with the naming schema of the OVAL files, which the Update Management uses to classify updates during the assessment. This prevents Update Management from properly matching classifications based on filtering rules during the assessment of missing updates.

This doesn't affect the deployment of updates. As a different logic is used in security update assessments, results might differ from the security updates applied during deployment. If you have classification set as **Critical** and **Security**, the update deployment will function as expected. Only the *classification of updates* during an assessment is affected.

**Update Management for Windows Server machines is unaffected; update classification and deployments are unchanged.**

## Integrate Update Management with Configuration Manager

Customers who have invested in Microsoft Endpoint Configuration Manager for managing PCs, servers, and mobile devices also rely on the strength and maturity of Configuration Manager to help manage software updates. To learn how to integrate Update Management with Configuration Manager, see [Integrate Update Management with Windows Endpoint Configuration Manager](#).

## Third-party updates on Windows

Update Management relies on the locally configured update repository to update supported Windows systems, either WSUS or Windows Update. Tools such as [System Center Updates Publisher](#) allow you to import and publish custom updates with WSUS. This scenario allows Update Management to update machines that use Configuration Manager as their update repository with third-party software. To learn how to configure Updates Publisher, see [Install Updates Publisher](#).

## Next steps

- Before enabling and using Update Management, review [Plan your Update Management deployment](#).
- Review commonly asked questions about Update Management in the [Azure Automation frequently asked questions](#).

# Maintenance for virtual machines in Azure

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure periodically updates its platform to improve the reliability, performance, and security of the host infrastructure for virtual machines. The purpose of these updates ranges from patching software components in the hosting environment to upgrading networking components or decommissioning hardware.

Updates rarely affect the hosted VMs. When updates do have an effect, Azure chooses the least impactful method for updates:

- If the update doesn't require a reboot, the VM is paused while the host is updated, or the VM is live-migrated to an already updated host.
- If maintenance requires a reboot, you're notified of the planned maintenance. Azure also provides a time window in which you can start the maintenance yourself, at a time that works for you. The self-maintenance window is typically 35 days (for Host machines) unless the maintenance is urgent. Azure is investing in technologies to reduce the number of cases in which planned platform maintenance requires the VMs to be rebooted. For instructions on managing planned maintenance, see [Handling planned maintenance notifications using the Azure CLI, PowerShell or portal](#).

This page describes how Azure performs both types of maintenance. For more information about unplanned events (outages), see [Manage the availability of VMs for Windows](#) or the corresponding article for [Linux](#).

Within a VM, you can get notifications about upcoming maintenance by [using Scheduled Events for Windows](#) or for [Linux](#).

## Maintenance that doesn't require a reboot

Most platform updates don't affect customer VMs. When a no-impact update isn't possible, Azure chooses the update mechanism that's least impactful to customer VMs.

Most nonzero-impact maintenance pauses the VM for less than 10 seconds. In certain cases, Azure uses memory-preserving maintenance mechanisms. These mechanisms pause the VM, typically for about 30 seconds, and preserve the memory in RAM. The VM is then resumed, and its clock is automatically synchronized.

Memory-preserving maintenance works for more than 90 percent of Azure VMs. It doesn't work for G, L, M, N, and H series. Azure increasingly uses live-migration technologies and improves memory-preserving maintenance mechanisms to reduce the pause durations.

These maintenance operations that don't require a reboot are applied one fault domain at a time. They stop if they receive any warning health signals from platform monitoring tools. Maintenance operations that do not require a reboot may occur simultaneously in paired regions or Availability Zones. For a given change, the deployment are mostly sequenced across Availability Zones and across Region pairs, but there can be overlap at the tail.

These types of updates can affect some applications. When the VM is live-migrated to a different host, some sensitive workloads might show a slight performance degradation in the few minutes leading up to the VM pause. To prepare for VM maintenance and reduce impact during Azure maintenance, try [using Scheduled Events for Windows](#) or [Linux](#) for such applications.

For greater control on all maintenance activities including zero-impact and rebootless updates, you can create a

Maintenance Configuration feature. Creating a Maintenance Configuration gives you the option to skip all platform updates and apply the updates at your choice of time. For more information, see [Managing platform updates with Maintenance Configurations](#).

## Live migration

Live migration is an operation that doesn't require a reboot and that preserves memory for the VM. It causes a pause or freeze, typically lasting no more than 5 seconds. Except for G, M, N, and H series, all infrastructure as a service (IaaS) VMs, are eligible for live migration. Eligible VMs represent more than 90 percent of the IaaS VMs that are deployed to the Azure fleet.

### NOTE

You won't receive a notification in the Azure portal for live migration operations that don't require a reboot. To see a list of live migrations that don't require a reboot, [query for scheduled events](#).

The Azure platform starts live migration in the following scenarios:

- Planned maintenance
- Hardware failure
- Allocation optimizations

Some planned-maintenance scenarios use live migration, and you can use Scheduled Events to know in advance when live migration operations will start.

Live migration can also be used to move VMs when Azure Machine Learning algorithms predict an impending hardware failure or when you want to optimize VM allocations. For more information about predictive modeling that detects instances of degraded hardware, see [Improving Azure VM resiliency with predictive machine learning and live migration](#). Live-migration notifications appear in the Azure portal in the Monitor and Service Health logs as well as in Scheduled Events if you use these services.

## Maintenance that requires a reboot

In the rare case where VMs need to be rebooted for planned maintenance, you'll be notified in advance. Planned maintenance has two phases: the self-service phase and a scheduled maintenance phase.

During the *self-service phase*, which typically lasts four weeks, you start the maintenance on your VMs. As part of the self-service, you can query each VM to see its status and the result of your last maintenance request.

### NOTE

For VM-series that do not support [Live Migration](#), local (ephemeral) disks data can be lost during the maintenance events. See each individual VM-series for information on if Live Migration is supported.

When you start self-service maintenance, your VM is redeployed to an already updated node. Because the VM is redeployed, the temporary disk is lost and dynamic IP addresses associated with the virtual network interface are updated.

If an error arises during self-service maintenance, the operation stops, the VM isn't updated, and you get the option to retry the self-service maintenance.

When the self-service phase ends, the *scheduled maintenance phase* begins. During this phase, you can still query for the maintenance phase, but you can't start the maintenance yourself.

For more information on managing maintenance that requires a reboot, see [Handling planned maintenance notifications](#) using the Azure [CLI](#), [PowerShell](#) or [portal](#).

## Availability considerations during scheduled maintenance

If you decide to wait until the scheduled maintenance phase, there are a few things you should consider to maintain the highest availability of your VMs.

### Paired regions

Each Azure region is paired with another region within the same geographical vicinity. Together, they make a region pair. During the scheduled maintenance phase, Azure updates only the VMs in a single region of a region pair. For example, while updating the VM in North Central US, Azure doesn't update any VM in South Central US at the same time. However, other regions such as North Europe can be under maintenance at the same time as East US. Understanding how region pairs work can help you better distribute your VMs across regions. For more information, see [Azure region pairs](#).

### Availability zones

Availability zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions.

An availability zone is a combination of a fault domain and an update domain. If you create three or more VMs across three zones in an Azure region, your VMs are effectively distributed across three fault domains and three update domains. The Azure platform recognizes this distribution across update domains to make sure that VMs in different zones are not updated at the same time.

Each infrastructure update rolls out zone by zone, within a single region. But, you can have deployment going on in Zone 1, and different deployment going in Zone 2, at the same time. Deployments are not all serialized. But, a single deployment that requires a reboot only rolls out one zone at a time to reduce risk. In general, updates that require a reboot are avoided when possible, and Azure attempts to use Live Migration or provide customers control.

### Virtual machine scale sets

Virtual machine scale sets in **Flexible** orchestration mode are an Azure compute resource allow you to combine the scalability of virtual machine scale sets in **Uniform** orchestration mode with the regional availability guarantees of availability sets.

With Flexible orchestration, you can choose whether your instances are spread across multiple zones, or spread across fault domains within a single region.

### Availability sets and Uniform scale sets

When deploying a workload on Azure VMs, you can create the VMs within an *availability set* to provide high availability to your application. Using availability sets, you can ensure that during either an outage or maintenance events that require a reboot, at least one VM is available.

Within an availability set, individual VMs are spread across up to 20 update domains. During scheduled maintenance, only one update domain is updated at any given time. Update domains aren't necessarily updated sequentially.

Virtual machine *scale sets* in **Uniform** orchestration mode are an Azure compute resource that you can use to deploy and manage a set of identical VMs as a single resource. The scale set is automatically deployed across UDs, like VMs in an availability set. As with availability sets, when you use Uniform scale sets, only one UD is updated at any given time during scheduled maintenance.

For more information about setting up your VMs for high availability, see [Manage the availability of your VMs for Windows](#) or the corresponding article for [Linux](#).

## Next steps

You can use the [Azure CLI](#), [Azure PowerShell](#), or the [portal](#) to manage planned maintenance.

# Handling planned maintenance notifications

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Azure periodically performs updates to improve the reliability, performance, and security of the host infrastructure for virtual machines. Updates are changes like patching the hosting environment or upgrading and decommissioning hardware. A majority of these updates are completed without any impact to the hosted virtual machines. However, there are cases where updates do have an impact:

- If the maintenance does not require a reboot, Azure pauses the VM for few seconds while the host is updated. These types of maintenance operations are applied fault domain by fault domain. Progress is stopped if any warning health signals are received.
- If maintenance requires a reboot, you get a notice of when the maintenance is planned. You are given a time window of about 35 days where you can start the maintenance yourself, when it works for you.

Planned maintenance that requires a reboot is scheduled in waves. Each wave has different scope (regions).

- A wave starts with a notification to customers. By default, the notification is sent to the subscription admin and co-admins. You can add more recipients and messaging options like email, SMS, and webhooks, using [Activity Log Alerts](#).
- Once a notification goes out, a *self-service window* is made available. During this window, you can query which of your virtual machines are affected and start maintenance based on your own scheduling needs. The self-service window is typically about 35 days.
- After the self-service window, a *scheduled maintenance window* begins. At some point during this window, Azure schedules and applies the required maintenance to your virtual machine.

The goal in having two windows is to give you enough time to start maintenance and reboot your virtual machine while knowing when Azure will automatically start maintenance.

You can use the Azure portal, PowerShell, REST API, and CLI to query for the maintenance windows for your VMs and start self-service maintenance.

## Should you start maintenance using during the self-service window?

The following guidelines should help you decide whether to use this capability and start maintenance at your own time.

### NOTE

Self-service maintenance might not be available for all of your VMs. To determine if proactive redeploy is available for your VM, look for the **Start now** in the maintenance status. Self-service maintenance is currently not available for Cloud Services (Web/Worker Role) and Service Fabric.

Self-service maintenance is not recommended for deployments using **availability sets**. Availability sets are already only updated one update domain at a time.

- Let Azure trigger the maintenance. For maintenance that requires reboot, maintenance will be done update domain by update domain. The update domains do not necessarily receive the maintenance sequentially, and that there is a 30-minute pause between update domains.
- If a temporary loss of some capacity (1 update domain) is a concern, you can add instances during the

maintenance period.

- For maintenance that does not require reboot, updates are applied at the fault domain level.

**Don't** use self-service maintenance in the following scenarios:

- If you shut down your VMs frequently, either manually, using DevTest Labs, using auto-shutdown, or following a schedule, it could revert the maintenance status and therefore cause additional downtime.
- On short-lived VMs that you know will be deleted before the end of the maintenance wave.
- For workloads with a large state stored in the local (ephemeral) disk that is desired to be maintained upon update.
- For cases where you resize your VM often, as it could revert the maintenance status.
- If you have adopted scheduled events that enable proactive failover or graceful shutdown of your workload, 15 minutes before start of maintenance shutdown

**Use** self-service maintenance, if you are planning to run your VM uninterrupted during the scheduled maintenance phase and none of the counter-indications mentioned above are applicable.

It is best to use self-service maintenance in the following cases:

- You need to communicate an exact maintenance window to your management or end-customer.
- You need to complete the maintenance by a given date.
- You need to control the sequence of maintenance, for example, multi-tier application to guarantee safe recovery.
- More than 30 minutes of VM recovery time is needed between two update domains (UDs). To control the time between update domains, you must trigger maintenance on your VMs one update domain (UD) at a time.

## FAQ

**Q: Why do you need to reboot my virtual machines now?**

**A:** While the majority of updates and upgrades to the Azure platform do not impact virtual machine's availability, there are cases where we can't avoid rebooting virtual machines hosted in Azure. We have accumulated several changes that require us to restart our servers that will result in virtual machines reboot.

**Q: If I follow your recommendations for High Availability by using an Availability Set, am I safe?**

**A:** Virtual machines deployed in an availability set or virtual machine scale sets have the notion of Update Domains (UD). When performing maintenance, Azure honors the UD constraint and will not reboot virtual machines from different UD (within the same availability set). Azure also waits for at least 30 minutes before moving to the next group of virtual machines.

For more information about high availability, see [Availability for virtual machines in Azure](#).

**Q: How do I get notified about planned maintenance?**

**A:** A planned maintenance wave starts by setting a schedule to one or more Azure regions. Soon after, an email notification is sent to the subscription admins, co-admins, owners, and contributors (One email per subscription with all recipients added). Additional channels and recipients for this notification could be configured using Activity Log Alerts. In case you deploy a virtual machine to a region where planned maintenance is already scheduled, you will not receive the notification but rather need to check the maintenance state of the VM.

**Q: I don't see any indication of planned maintenance in the portal, PowerShell, or CLI. What is wrong?**

**A:** Information related to planned maintenance is available during a planned maintenance wave only for the VMs that are going to be impacted by it. In other words, if you see no data, it could be that the maintenance

wave has already completed (or not started) or that your virtual machine is already hosted in an updated server.

**Q: Is there a way to know exactly when my virtual machine will be impacted?**

**A:** When setting the schedule, we define a time window of several days. However, the exact sequencing of servers (and VMs) within this window is unknown. Customers who would like to know the exact time for their VMs can use [scheduled events](#) and query from within the virtual machine and receive a 15-minute notification before a VM reboot.

**Q: How long will it take you to reboot my virtual machine?**

**A:** Depending on the size of your VM, reboot may take up to several minutes during the self-service maintenance window. During the Azure initiated reboots in the scheduled maintenance window, the reboot will typically take about 25 minutes. Note that in case you use Cloud Services (Web/Worker Role), Virtual Machine Scale Sets, or availability sets, you will be given 30 minutes between each group of VMs (UD) during the scheduled maintenance window.

**Q: What is the experience in the case of Virtual Machine Scale Sets?**

**A:** Planned maintenance is now available for Virtual Machine Scale Sets. For instructions on how to initiate self-service maintenance refer [planned maintenance for virtual machine scale sets](#) document.

**Q: What is the experience in the case of Cloud Services (Web/Worker Role) and Service Fabric?**

**A:** While these platforms are impacted by planned maintenance, customers using these platforms are considered safe given that only VMs in a single Upgrade Domain (UD) will be impacted at any given time. Self-service maintenance is currently not available for Cloud Services (Web/Worker Role) and Service Fabric.

**Q: I don't see any maintenance information on my VMs. What went wrong?**

**A:** There are several reasons why you're not seeing any maintenance information on your VMs:

1. You are using a subscription marked as Microsoft internal.
2. Your VMs are not scheduled for maintenance. It could be that the maintenance wave has ended, canceled, or modified so that your VMs are no longer impacted by it.
3. You have deallocated VM and then started it. This can cause VM to move to a location which does not have planned maintenance wave scheduled. So the VM will not show maintenance information any more.
4. You don't have the **Maintenance** column added to your VM list view. While we have added this column to the default view, customers who configured to see non-default columns must manually add the **Maintenance** column to their VM list view.

**Q: My VM is scheduled for maintenance for the second time. Why?**

**A:** There are several use cases where you will see your VM scheduled for maintenance after you have already completed your maintenance-redeploy:

1. We have canceled the maintenance wave and restarted it with a different payload. It could be that we've detected faulted payload and we simply need to deploy an additional payload.
2. Your VM was *service healed* to another node due to a hardware fault.
3. You have selected to stop (deallocate) and restart the VM.
4. You have **auto shutdown** turned on for the VM.

## Next steps

You can handle planned maintenance using the [Azure CLI](#), [Azure PowerShell](#) or [portal](#).

# Handling planned maintenance notifications using the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can use the CLI to see when VMs are scheduled for [maintenance](#). Planned maintenance information is available from [az vm get-instance-view](#).

Maintenance information is returned only if there is maintenance planned.

```
az vm get-instance-view -n myVM -g myResourceGroup --query instanceView.maintenanceRedeployStatus
```

Output

```
"maintenanceRedeployStatus": {  
    "additionalProperties": {},  
    "isCustomerInitiatedMaintenanceAllowed": true,  
    "lastOperationMessage": null,  
    "lastOperationResultCode": "None",  
    "maintenanceWindowEndTime": "2018-06-04T16:30:00+00:00",  
    "maintenanceWindowStartTime": "2018-05-21T16:30:00+00:00",  
    "preMaintenanceWindowEndTime": "2018-05-19T12:30:00+00:00",  
    "preMaintenanceWindowStartTime": "2018-05-14T12:30:00+00:00"
```

## Start maintenance

The following call will start maintenance on a VM if `IsCustomerInitiatedMaintenanceAllowed` is set to true.

```
az vm perform-maintenance -g myResourceGroup -n myVM
```

## Classic deployments

### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

If you still have legacy VMs that were deployed using the classic deployment model, you can use the Azure classic CLI to query for VMs and initiate maintenance.

Make sure you are in the correct mode to work with classic VM by typing:

```
azure config mode asm
```

To get the maintenance status of a VM named *myVM*, type:

```
azure vm show myVM
```

To start maintenance on your classic VM named *myVM* in the *myService* service and *myDeployment* deployment, type:

```
azure compute virtual-machine initiate-maintenance --service-name myService --name myDeployment --virtual-machine-name myVM
```

## Next steps

You can also handle planned maintenance using the [Azure PowerShell](#) or [portal](#).

# Handling planned maintenance notifications using the portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Once a [planned maintenance](#) wave is scheduled, you can check for a list of virtual machines that are impacted.

You can use the Azure portal and look for VMs scheduled for maintenance.

1. Sign in to the [Azure portal](#).
2. In the left navigation, click **Virtual Machines**.
3. In the Virtual Machines pane, select **Maintenance -> Virtual machine maintenance** button to open the list with maintenance columns.

**Maintenance status:** Shows the maintenance status for the VM. The following are the potential values:

VALUE	DESCRIPTION
Start now	The VM is in the self-service maintenance window that lets you initiate the maintenance yourself. See below on how to start maintenance on your VM.
Scheduled	The VM is scheduled for maintenance with no option for you to initiate maintenance. You can learn of the maintenance window by selecting the Maintenance - Scheduled window in this view or by clicking on the VM.
Already updated	Your VM is already updated and no further action is required at this time.
Retry later	You have initiated maintenance with no success. You will be able to use the self-service maintenance option at a later time.
Retry now	You can retry a previously unsuccessful self-initiated maintenance.
-	Your VM is not part of a planned maintenance wave.

**Maintenance - Self-service window:** Shows the time window when you can self-start maintenance on your VMs.

**Maintenance - Scheduled window:** Shows the time window when Azure will maintain your VM in order to complete maintenance.

## Notification and alerts in the portal

Azure communicates a schedule for planned maintenance by sending an email to the subscription owner and co-owners group. You can add additional recipients and channels to this communication by creating Azure activity log alerts. For more information, see [Create activity log alerts on service notifications](#).

Make sure you set the **Event type** as **Planned maintenance**, and **Services** as **Virtual Machine Scale Sets** and/or **Virtual Machines**.

## Start Maintenance on your VM from the portal

While looking at the VM details, you will be able to see more maintenance-related details.

At the top of the VM details view, a new notification ribbon will be added if your VM is included in a planned maintenance wave. In addition, a new option is added to start maintenance when possible.

Click on the maintenance notification to see the maintenance page with more details on the planned maintenance. From there, you will be able to **start maintenance** on your VM.

Once you start maintenance, your virtual machine will be maintained and the maintenance status will be updated to reflect the result within few minutes.

If you missed the self-service window, you will still be able to see the window when your VM will be maintained by Azure.

## Next steps

You can also handle planned maintenance using the [Azure CLI](#) or [PowerShell](#).

# Handling planned maintenance using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

You can use Azure PowerShell to see when VMs are scheduled for [maintenance](#). Planned maintenance information is available from the [Get-AzVM](#) cmdlet when you use the `-status` parameter.

Maintenance information is returned only if there is maintenance planned. If no maintenance is scheduled that impacts the VM, the cmdlet does not return any maintenance information.

```
Get-AzVM -ResourceGroupName myResourceGroup -Name myVM -Status
```

## Output

```
MaintenanceRedeployStatus      :  
IsCustomerInitiatedMaintenanceAllowed : True  
PreMaintenanceWindowStartTime    : 5/14/2018 12:30:00 PM  
PreMaintenanceWindowEndTime     : 5/19/2018 12:30:00 PM  
MaintenanceWindowStartTime      : 5/21/2018 4:30:00 PM  
MaintenanceWindowEndTime        : 6/4/2018 4:30  
LastOperationResultCode         : None
```

The following properties are returned under MaintenanceRedeployStatus:

VALUE	DESCRIPTION
IsCustomerInitiatedMaintenanceAllowed	Indicates whether you can start maintenance on the VM at this time
PreMaintenanceWindowStartTime	The beginning of the maintenance self-service window when you can initiate maintenance on your VM
PreMaintenanceWindowEndTime	The end of the maintenance self-service window when you can initiate maintenance on your VM
MaintenanceWindowStartTime	The beginning of the maintenance scheduled in which Azure initiates maintenance on your VM
MaintenanceWindowEndTime	The end of the maintenance scheduled window in which Azure initiates maintenance on your VM
LastOperationResultCode	The result of the last attempt to initiate maintenance on the VM

You can also get the maintenance status for all VMs in a resource group by using [Get-AzVM](#) and not specifying a VM.

```
Get-AzVM -ResourceGroupName myResourceGroup -Status
```

The following PowerShell example takes your subscription ID and returns a list of VMs that are scheduled for

maintenance.

```
function MaintenanceIterator
{
    Select-AzSubscription -SubscriptionId $args[0]

    $rgList= Get-AzResourceGroup

    for ($rgIdx=0; $rgIdx -lt $rgList.Length ; $rgIdx++)
    {
        $rg = $rgList[$rgIdx]
        $vmList = Get-AzVM -ResourceGroupName $rg.ResourceGroupName
        for ($vmIdx=0; $vmIdx -lt $vmList.Length ; $vmIdx++)
        {
            $vm = $vmList[$vmIdx]
            $vmDetails = Get-AzVM -ResourceGroupName $rg.ResourceGroupName -Name $vm.Name -Status
            if ($vmDetails.MaintenanceRedeployStatus )
            {
                Write-Output "VM: $($vmDetails.Name) IsCustomerInitiatedMaintenanceAllowed:
$($vmDetails.MaintenanceRedeployStatus.IsCustomerInitiatedMaintenanceAllowed)
 $($vmDetails.MaintenanceRedeployStatus.LastOperationMessage)"
            }
        }
    }
}
```

## Start maintenance on your VM using PowerShell

Using information from the function in the previous section, the following starts maintenance on a VM if **IsCustomerInitiatedMaintenanceAllowed** is set to true.

```
Restart-AzVM -PerformMaintenance -name $vm.Name -ResourceGroupName $rg.ResourceGroupName
```

## Classic deployments

### IMPORTANT

VMs created through the classic deployment model will be retired on March 1, 2023.

If you use IaaS resources from Azure Service Management, please complete your migration by March 1, 2023. We encourage you to make the switch sooner to take advantage of the many feature enhancements in Azure Resource Manager.

For more information, see [Migrate your IaaS resources to Azure Resource Manager by March 1, 2023](#).

If you still have legacy VMs that were deployed using the classic deployment model, you can use PowerShell to query for VMs and initiate maintenance.

To get the maintenance status of a VM, type:

```
Get-AzureVM -ServiceName <Service name> -Name <VM name>
```

To start maintenance on your classic VM, type:

```
Restart-AzureVM -InitiateMaintenance -ServiceName <service name> -Name <VM name>
```

## Next steps

You can also handle planned maintenance using the [Azure CLI](#) or [portal](#).

# Managing VM updates with Maintenance Configurations

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Maintenance Configurations give you the ability to control and manage updates for many Azure virtual machine resources since Azure frequently updates its infrastructure to improve reliability, performance, security or launch new features. Most updates are transparent to users, but some sensitive workloads, like gaming, media streaming, and financial transactions, can't tolerate even few seconds of a VM freezing or disconnecting for maintenance. Maintenance configurations is integrated with Azure Resource Graph (ARG) for low latency and high scale customer experience.

## IMPORTANT

Users are required to have a role of at least contributor in order to use maintenance configurations.

## Scopes

Maintenance Configurations currently supports three (3) scopes: Host, OS image, and Guest. While each scope allows scheduling and managing updates, the major difference lies in the resource they each support. This section outlines the details on the various scopes and their supported types:

SCOPE	SUPPORT RESOURCES
Host	Isolated Virtual Machines, Isolated Virtual Machine Scale Sets, Dedicated Hosts
OS Image	Virtual Machine Scale Sets
Guest	Virtual Machines, Azure Arc Servers

### Host

With this scope, you can manage platform updates that do not require a reboot on your *isolated VMs*, *isolated Virtual Machine Scale Set instances* and *dedicated hosts*. Some features and limitations unique to the host scope are:

- Schedules can be set anytime within 35 days. After 35 days, updates are automatically applied.
- A minimum of a 2 hour maintenance window is required for this scope.

[Learn more about Azure Dedicated Hosts](#)

### OS image

Using this scope with maintenance configurations lets you decide when to apply upgrades to OS disks in your *virtual machine scale sets* through an easier and more predictable experience. An upgrade works by replacing the OS disk of a VM with a new disk created using the latest image version. Any configured extensions and custom data scripts are run on the OS disk, while data disks are retained. Some features and limitations unique to this scope are:

- Scale sets need to have [automatic OS upgrades](#) enabled in order to use maintenance configurations.
- Schedule recurrence is defaulted to daily
- A minimum of 5 hours is required for the maintenance window

## Guest

This scope is integrated with [update management center](#) which allows you to save recurring deployment schedules to install updates for your Windows Server and Linux machines in Azure, in on-premises environments, and in other cloud environments connected using Azure Arc-enabled servers. Some features and limitations unique to this scope include:

- [Patch orchestration](#) for virtual machines need to be set to AutomaticByPlatform
- A minimum of 1 hour and 10 minutes is required for the maintenance window.
- There is no limit to the recurrence of your schedule

To learn more about this topic, checkout [update management center and scheduled patching](#)

## Management options

You can create and manage maintenance configurations using any of the following options:

- [Azure CLI](#)
- [Azure PowerShell](#)
- [Azure portal](#)

For an Azure Functions sample, see [Scheduling Maintenance Updates with Maintenance Configurations and Azure Functions](#).

## Next steps

To learn more, see [Maintenance and updates](#).

# Control updates with Maintenance Configurations and the Azure CLI

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Maintenance Configurations lets you decide when to apply platform updates to various Azure resources. This topic covers the Azure CLI options for Dedicated Hosts and Isolated VMs. For more about benefits of using Maintenance Configurations, its limitations, and other management options, see [Managing platform updates with Maintenance Configurations](#).

## IMPORTANT

There are different **scopes** which support certain machine types and schedules, so please ensure you are selecting the right scope for your virtual machine.

## Create a maintenance configuration

Use `az maintenance configuration create` to create a maintenance configuration. This example creates a maintenance configuration named *myConfig* scoped to the host.

```
az group create \
--location eastus \
--name myMaintenanceRG
az maintenance configuration create \
-g myMaintenanceRG \
--resource-name myConfig \
--maintenance-scope host\
--location eastus
```

Copy the configuration ID from the output to use later.

Using `--maintenance-scope host` ensures that the maintenance configuration is used for controlling updates to the host infrastructure.

If you try to create a configuration with the same name, but in a different location, you will get an error. Configuration names must be unique to your resource group.

You can query for available maintenance configurations using `az maintenance configuration list`.

```
az maintenance configuration list --query "[].{Name:name, ID:id}" -o table
```

## Create a maintenance configuration with scheduled window

You can also declare a scheduled window when Azure will apply the updates on your resources. This example creates a maintenance configuration named *myConfig* with a scheduled window of 5 hours on the fourth Monday of every month. Once you create a scheduled window you no longer have to apply the updates manually.

```
az maintenance configuration create \
-g myMaintenanceRG \
--resource-name myConfig \
--maintenance-scope host \
--location eastus \
--maintenance-window-duration "05:00" \
--maintenance-window-recur-every "Month Fourth Monday" \
--maintenance-window-start-date-time "2020-12-30 08:00" \
--maintenance-window-time-zone "Pacific Standard Time"
```

#### IMPORTANT

Maintenance **duration** must be *2 hours* or longer.

Maintenance recurrence can be expressed as daily, weekly or monthly. Some examples are:

- **daily**- maintenance-window-recur-every: "Day" **or** "3Days"
- **weekly**- maintenance-window-recur-every: "3Weeks" **or** "Week Saturday,Sunday"
- **monthly**- maintenance-window-recur-every: "Month day23,day24" **or** "Month Last Sunday" **or** "Month Fourth Monday"

## Assign the configuration

Use `az maintenance assignment create` to assign the configuration to your machine.

#### Isolated VM

Apply the configuration to a VM using the ID of the configuration. Specify `--resource-type virtualMachines` and supply the name of the VM for `--resource-name`, and the resource group for to the VM in `--resource-group`, and the location of the VM for `--location`.

```
az maintenance assignment create \
--resource-group myMaintenanceRG \
--location eastus \
--resource-name myVM \
--resource-type virtualMachines \
--provider-name Microsoft.Compute \
--configuration-assignment-name myConfig \
--maintenance-configuration-id "/subscriptions/1111abcd-1a11-1a2b-1a12-123456789abc/resourcegroups/myMaintenanceRG/providers/Microsoft.Maintenance/maintenanceConfigurations/myConfig"
```

#### Dedicated host

To apply a configuration to a dedicated host, you need to include `--resource-type hosts`, `--resource-parent-name` with the name of the host group, and `--resource-parent-type hostGroups`.

The parameter `--resource-id` is the ID of the host. You can use [az-vm-host-get-instance-view](#) to get the ID of your dedicated host.

```
az maintenance assignment create \
    -g myDHResourceGroup \
    --resource-name myHost \
    --resource-type hosts \
    --provider-name Microsoft.Compute \
    --configuration-assignment-name myConfig \
    --maintenance-configuration-id "/subscriptions/1111abcd-1a11-1a2b-1a12-
123456789abc/resourcegroups/myDHResourceGroup/providers/Microsoft.Maintenance/maintenanceConfigurations/myCo
nfig" \
    -l eastus \
    --resource-parent-name myHostGroup \
    --resource-parent-type hostGroups
```

## Check configuration

You can verify that the configuration was applied correctly, or check to see what configuration is currently applied using `az maintenance assignment list`.

### Isolated VM

```
az maintenance assignment list \
    --provider-name Microsoft.Compute \
    --resource-group myMaintenanceRG \
    --resource-name myVM \
    --resource-type virtualMachines \
    --query "[].{resource:resourceGroup, configName:name}" \
    --output table
```

### Dedicated host

```
az maintenance assignment list \
    --resource-group myDHResourceGroup \
    --resource-name myHost \
    --resource-type hosts \
    --provider-name Microsoft.Compute \
    --resource-parent-name myHostGroup \
    --resource-parent-type hostGroups \
    --query "[].{ResourceGroup:resourceGroup,configName:name}" \
    -o table
```

## Check for pending updates

Use `az maintenance update list` to see if there are pending updates. Update `--subscription` to be the ID for the subscription that contains the VM.

If there are no updates, the command will return an error message, which will contain the text:

```
Resource not found...StatusCode: 404 .
```

If there are updates, only one will be returned, even if there are multiple updates pending. The data for this update will be returned in an object:

```
[  
 {  
   "impactDurationInSec": 9,  
   "impactType": "Freeze",  
   "maintenanceScope": "Host",  
   "notBefore": "2020-03-03T07:23:04.905538+00:00",  
   "resourceId": "/subscriptions/9120c5ff-e78e-4bd0-b29f-  
75c19cadd078/resourcegroups/DemoRG/providers/Microsoft.Compute/hostGroups/demoHostGroup/hosts/myHost",  
   "status": "Pending"  
 }  
]
```

## Isolated VM

Check for pending updates for an isolated VM. In this example, the output is formatted as a table for readability.

```
az maintenance update list \  
-g myMaintenanceRg \  
--resource-name myVM \  
--resource-type virtualMachines \  
--provider-name Microsoft.Compute \  
-o table
```

## Dedicated host

To check for pending updates for a dedicated host. In this example, the output is formatted as a table for readability. Replace the values for the resources with your own.

```
az maintenance update list \  
--subscription 1111abcd-1a11-1a2b-1a12-123456789abc \  
-g myHostResourceGroup \  
--resource-name myHost \  
--resource-type hosts \  
--provider-name Microsoft.Compute \  
--resource-parentname myHostGroup \  
--resource-parent-type hostGroups \  
-o table
```

## Apply updates

Use `az maintenance apply update` to apply pending updates. On success, this command will return JSON containing the details of the update. Apply update calls can take upto 2 hours to complete.

### Isolated VM

Create a request to apply updates to an isolated VM.

```
az maintenance applyupdate create \  
--subscription 1111abcd-1a11-1a2b-1a12-123456789abc \  
--resource-group myMaintenanceRG \  
--resource-name myVM \  
--resource-type virtualMachines \  
--provider-name Microsoft.Compute
```

### Dedicated host

Apply updates to a dedicated host.

```
az maintenance applyupdate create \
--subscription 1111abcd-1a11-1a2b-1a12-123456789abc \
--resource-group myHostResourceGroup \
--resource-name myHost \
--resource-type hosts \
--provider-name Microsoft.Compute \
--resource-parent-name myHostGroup \
--resource-parent-type hostGroups
```

## Check the status of applying updates

You can check on the progress of the updates using `az maintenance applyupdate get`.

You can use `default` as the update name to see results for the last update, or replace `myUpdateName` with the name of the update that was returned when you ran `az maintenance applyupdate create`.

```
Status      : Completed
ResourceId   : /subscriptions/12ae7457-4a34-465c-94c1-
               17c058c2bd25/resourcegroups/TestShants/providers/Microsoft.Com-
               pute/virtualMachines/DXT-test-04-iso
LastUpdateTime : 1/1/2020 12:00:00 AM
Id          : /subscriptions/12ae7457-4a34-465c-94c1-
               17c058c2bd25/resourcegroups/TestShants/providers/Microsoft.Com-
               pute/virtualMachines/DXT-test-04-iso/providers/Microsoft.Mainten-
               ance/applyUpdates/default
Name        : default
Type        : Microsoft.Maintenance/applyUpdates
```

LastUpdateTime will be the time when the update got complete, either initiated by you or by the platform in case self-maintenance window was not used. If there has never been an update applied through maintenance control it will show default value.

### Isolated VM

```
az maintenance applyupdate get \
--resource-group myMaintenanceRG \
--resource-name myVM \
--resource-type virtualMachines \
--provider-name Microsoft.Compute \
--apply-update-name default
```

### Dedicated host

```
az maintenance applyupdate get \
--subscription 1111abcd-1a11-1a2b-1a12-123456789abc \
--resource-group myMaintenanceRG \
--resource-name myHost \
--resource-type hosts \
--provider-name Microsoft.Compute \
--resource-parent-name myHostGroup \
--resource-parent-type hostGroups \
--apply-update-name myUpdateName \
--query "{LastUpdate:lastUpdateTime, Name:name, ResourceGroup:resourceGroup, Status:status}" \
--output table
```

## Delete a maintenance configuration

Use `az maintenance configuration delete` to delete a maintenance configuration. Deleting the configuration removes the maintenance control from the associated resources.

```
az maintenance configuration delete \
--subscription 1111abcd-1a11-1a2b-1a12-123456789abc \
-g myResourceGroup \
--resource-name myConfig
```

## Next steps

To learn more, see [Maintenance and updates](#).

# Control updates with Maintenance Configurations and Azure PowerShell

9/21/2022 • 5 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Creating a Maintenance Configurations lets you decide when to apply platform updates to various Azure resources. This topic covers the Azure PowerShell options for Dedicated Hosts and Isolated VMs. For more about benefits of using Maintenance Configurations, its limitations, and other management options, see [Managing platform updates with Maintenance Configurations](#).

If you are looking for information about Maintenance Configurations for scale sets, see [Maintenance Control for virtual machine scale sets](#).

## IMPORTANT

There are different **scopes** which support certain machine types and schedules, so please ensure you are selecting the right scope for your virtual machine.

## Enable the PowerShell module

Make sure `PowerShellGet` is up to date.

```
Install-Module -Name PowerShellGet -Repository PSGallery -Force
```

Install the `Az.Maintenance` PowerShell module.

```
Install-Module -Name Az.Maintenance
```

If you are installing locally, make sure you open your PowerShell prompt as an administrator.

You may also be asked to confirm that you want to install from an *untrusted repository*. Type `y` or select **Yes to All** to install the module.

## Create a maintenance configuration

Create a resource group as a container for your configuration. In this example, a resource group named `myMaintenanceRG` is created in `eastus`. If you already have a resource group that you want to use, you can skip this part and replace the resource group name with your own in the rest of the examples.

```
New-AzResourceGroup  
-Location eastus  
-Name myMaintenanceRG
```

Use `New-AzMaintenanceConfiguration` to create a maintenance configuration. This example creates a maintenance configuration named `myConfig` scoped to the host.

```
$config = New-AzMaintenanceConfiguration `  
    -ResourceGroup myMaintenanceRG `  
    -Name myConfig `  
    -MaintenanceScope host `  
    -Location eastus
```

Using `-MaintenanceScope host` ensures that the maintenance configuration is used for controlling updates to the host.

If you try to create a configuration with the same name, but in a different location, you will get an error. Configuration names must be unique to your resource group.

You can query for available maintenance configurations using [Get-AzMaintenanceConfiguration](#).

```
Get-AzMaintenanceConfiguration | Format-Table -Property Name,Id
```

### Create a maintenance configuration with scheduled window

You can also declare a scheduled window when Azure will apply the updates on your resources. This example creates a maintenance configuration named myConfig with a scheduled window of 5 hours on the fourth Monday of every month. Once you create a scheduled window you no longer have to apply the updates manually.

```
$config = New-AzMaintenanceConfiguration `  
    -ResourceGroup $RGName `  
    -Name $MaintenanceConfig `  
    -MaintenanceScope Host `  
    -Location $location `  
    -StartTime "2020-10-01 00:00" `  
    -TimeZone "Pacific Standard Time" `  
    -Duration "05:00" `  
    -RecurEvery "Month Fourth Monday"
```

#### IMPORTANT

Maintenance **duration** must be *2 hours* or longer.

Maintenance **recurrence** can be expressed as daily, weekly or monthly. Some examples are:

- **daily**- RecurEvery "Day" **or** "3Days"
- **weekly**- RecurEvery "3Weeks" **or** "Week Saturday,Sunday"
- **monthly**- RecurEvery "Month day23,day24" **or** "Month Last Sunday" **or** "Month Fourth Monday"

## Assign the configuration

Use [New-AzConfigurationAssignment](#) to assign the configuration to your isolated VM or Azure Dedicated Host.

### Isolated VM

Apply the configuration to a VM using the ID of the configuration. Specify `-ResourceType VirtualMachines` and supply the name of the VM for `-ResourceName`, and the resource group of the VM for `-ResourceGroupName`.

```
New-AzConfigurationAssignment ` 
-ResourceGroupName myResourceGroup ` 
-Location eastus ` 
-ResourceName myVM ` 
-ResourceType VirtualMachines ` 
-ProviderName Microsoft.Compute ` 
-ConfigurationAssignmentName $config.Name ` 
-MaintenanceConfigurationId $config.Id
```

## Dedicated host

To apply a configuration to a dedicated host, you also need to include `-ResourceType hosts`, `-ResourceParentName` with the name of the host group, and `-ResourceParentType hostGroups`.

```
New-AzConfigurationAssignment ` 
-ResourceGroupName myResourceGroup ` 
-Location eastus ` 
-ResourceName myHost ` 
-ResourceType hosts ` 
-ResourceParentName myHostGroup ` 
-ResourceParentType hostGroups ` 
-ProviderName Microsoft.Compute ` 
-ConfigurationAssignmentName $config.Name ` 
-MaintenanceConfigurationId $config.Id
```

## Check for pending updates

Use [Get-AzMaintenanceUpdate](#) to see if there are pending updates. Use `-subscription` to specify the Azure subscription of the VM if it is different from the one that you are logged into.

If there are no updates to show, this command will return nothing. Otherwise, it will return a `PSApplyUpdate` object:

```
{
  "maintenanceScope": "Host",
  "impactType": "Freeze",
  "status": "Pending",
  "impactDurationInSec": 9,
  "notBefore": "2020-02-21T16:47:44.8728029Z",
  "properties": {
    "resourceId": "/subscriptions/39c6cced-4d6c-4dd5-af86-57499cd3f846/resourcegroups/Ignite2019/providers/Microsoft.Compute/virtualMachines/MCDemo3"
  }
}
```

## Isolated VM

Check for pending updates for an isolated VM. In this example, the output is formatted as a table for readability.

```
Get-AzMaintenanceUpdate ` 
-ResourceGroupName myResourceGroup ` 
-ResourceName myVM ` 
-ResourceType VirtualMachines ` 
-ProviderName Microsoft.Compute | Format-Table
```

## Dedicated host

To check for pending updates for a dedicated host. In this example, the output is formatted as a table for readability. Replace the values for the resources with your own.

```
Get-AzMaintenanceUpdate ` 
-ResourceGroupName myResourceGroup ` 
-ResourceName myHost ` 
-ResourceType hosts ` 
-ResourceParentName myHostGroup ` 
-ResourceParentType hostGroups ` 
-ProviderName Microsoft.Compute | Format-Table
```

## Apply updates

Use [New-AzApplyUpdate](#) to apply pending updates. Apply update calls can take upto 2 hours to complete.

### Isolated VM

Create a request to apply updates to an isolated VM.

```
New-AzApplyUpdate ` 
-ResourceGroupName myResourceGroup ` 
-ResourceName myVM ` 
-ResourceType VirtualMachines ` 
-ProviderName Microsoft.Compute
```

On success, this command will return a `PSApplyUpdate` object. You can use the `Name` attribute in the [Get-AzApplyUpdate](#) command to check the update status. See [Check update status](#).

### Dedicated host

Apply updates to a dedicated host.

```
New-AzApplyUpdate ` 
-ResourceGroupName myResourceGroup ` 
-ResourceName myHost ` 
-ResourceType hosts ` 
-ResourceParentName myHostGroup ` 
-ResourceParentType hostGroups ` 
-ProviderName Microsoft.Compute
```

## Check update status

Use [Get-AzApplyUpdate](#) to check on the status of an update. The commands shown below show the status of the latest update by using `default` for the `-ApplyUpdateName` parameter. You can substitute the name of the update (returned by the [New-AzApplyUpdate](#) command) to get the status of a specific update.

```
Status      : Completed
ResourceId   : /subscriptions/12ae7457-4a34-465c-94c1-
17c058c2bd25/resourcegroups/TestShants/providers/Microsoft.Com-
pute/virtualMachines/DXT-test-04-iso
LastUpdateTime : 1/1/2020 12:00:00 AM
Id          : /subscriptions/12ae7457-4a34-465c-94c1-
17c058c2bd25/resourcegroups/TestShants/providers/Microsoft.Com-
pute/virtualMachines/DXT-test-04-iso/providers/Microsoft.Mainten-
ance/applyUpdates/default
Name        : default
Type        : Microsoft.Maintenance/applyUpdates
```

`LastUpdateTime` will be the time when the update got complete, either initiated by you or by the platform in case self-maintenance window was not used. If there has never been an update applied through maintenance configurations it will show `default` value.

## Isolated VM

Check for updates to a specific virtual machine.

```
Get-AzApplyUpdate ` 
    -ResourceGroupName myResourceGroup ` 
    -ResourceName myVM ` 
    -ResourceType VirtualMachines ` 
    -ProviderName Microsoft.Compute ` 
    -ApplyUpdateName default
```

## Dedicated host

Check for updates to a dedicated host.

```
Get-AzApplyUpdate ` 
    -ResourceGroupName myResourceGroup ` 
    -ResourceName myHost ` 
    -ResourceType hosts ` 
    -ResourceParentName myHostGroup ` 
    -ResourceParentType hostGroups ` 
    -ProviderName Microsoft.Compute ` 
    -ApplyUpdateName myUpdateName
```

## Remove a maintenance configuration

Use [Remove-AzMaintenanceConfiguration](#) to delete a maintenance configuration.

```
Remove-AzMaintenanceConfiguration ` 
    -ResourceGroupName myResourceGroup ` 
    -Name $config.Name
```

## Next steps

To learn more, see [Maintenance and updates](#).

# Control updates with Maintenance Configurations and the Azure portal

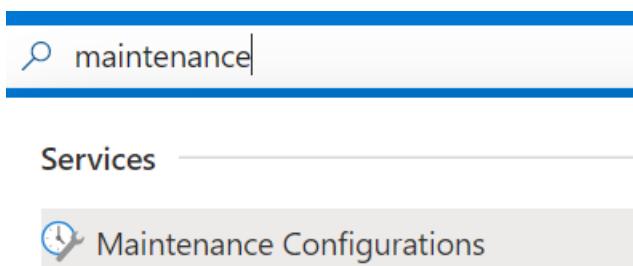
9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

With Maintenance Configurations, you can now take more control over when to apply updates to various Azure resources. This topic covers the Azure portal options for creating Maintenance Configurations. For more about benefits of using Maintenance Configurations, its limitations, and other management options, see [Managing platform updates with Maintenance Configurations](#).

## Create a Maintenance Configuration

1. Sign in to the Azure portal.
2. Search for **Maintenance Configurations**.



3. Click **Create**.

### Maintenance Configurations

Microsoft (microsoft.onmicrosoft.com)

[+ Create](#) [Manage view](#) [Refresh](#) [Export to CSV](#)

4. In the Basics tab, choose a subscription and resource group, provide a name for the configuration, choose a region, and select one of the scopes we offer which you wish to apply updates for. Click **Add a schedule** to add or modify the schedule for your configuration.

#### IMPORTANT

Certain virtual machine types and schedules will require a specific kind of scope. Check out [maintenance configuration scopes](#) to find the right one for your virtual machine.

## Create a maintenance configuration ...

Basics Machines Tags Review + create

Schedule your recurring updates by creating a maintenance configuration. Set your schedule first, then add your resources.  
[Learn more](#)

### Project details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	AISC-DEV-02
Resource group *	<input type="text"/> <a href="#">Create new</a>

### Instance details

Configuration name *	<input type="text"/>
Region *	(US) East US
Maintenance scope *	Host (isolated/dedicated hardware)
Schedule	<a href="#">Add a schedule</a>

5. In the Schedule tab, declare a scheduled window when Azure will apply the updates on your resources.

Set a start date, maintenance window, and recurrence if your resource requires it. Once you create a scheduled window you no longer have to apply the updates manually. Click **Next**.

#### IMPORTANT

Maintenance window duration must be *2 hours* or longer.

## Add/Modify schedule

X

Start on	<input type="text"/> MM/DD/YYYY <input type="button"/> h:mm A <input type="text"/> (UTC-08:00) Pacific Time (US & Canada) <input type="button"/>
Maintenance window (hours) *	<input type="text"/> 2
Repeats	<input type="text"/> 1 <input type="button"/> Day <input type="button"/>
Add end date	<input checked="" type="checkbox"/>
Ends on *	<input type="text"/> MM/DD/YYYY <input type="button"/> h:mm A
Schedule summary	Starts on: - Maintenance window: 2 hours Repeats: Does not repeat Ends on: -

6. In the Machines tab, assign resources now or skip this step and assign resources later after maintenance configuration deployment. Click **Next**.

7. Add tags and values. Click **Next**.

## Create a maintenance configuration

Basics Schedule Assignments Tags Review + create

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more about tags ↗](#)

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name ⓘ	Value ⓘ
project	: Infra
	:

8. Review the summary. Click **Create**.

9. After the deployment is complete, click **Go to resource**.

## Assign the configuration

On the details page of the maintenance configuration, click Machines and then click **Add Machine**.

The screenshot shows the Azure portal interface for managing a maintenance configuration named 'test'. The top navigation bar includes 'Home > test' and tabs for 'Machines', 'Schedule', 'Updates', 'Properties', and 'Locks'. On the left, there's a sidebar with 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', and 'Automation' sections. The main content area has a search bar, a 'Add machine' button (which is highlighted with a red box), and a 'Remove machines' button. A message below the buttons says: 'Manage resources assigned to this maintenance configuration. You can assign new resources, see the status of all resources currently assigned to the configuration, and by selecting resources below either apply pending maintenance or remove them from this configuration.' Below this message is a table with columns for 'Name ↑↓' and 'Type'. The 'Machines' section of the sidebar is currently selected.

Select the resources that you want the maintenance configuration assigned to and click **Ok**. The VM needs to be running to assign the configuration. An error occurs if you try to assign a configuration to a VM that is stopped.

Select resources

Select resources to assign to this maintenance configuration. Only resources that are not already assigned to a maintenance configuration are shown below. Currently, only dedicated hosts and isolated virtual machines are supported.

[Learn more about isolated virtual machines](#) [Learn more about dedicated hosts](#)

Filter by name...

Subscriptions: 12 of 24 selected – Don't see a subscription? [Open Directory + Subscription settings](#)

Subscription	All resource groups	All locations	All	
12 subscriptions	All resource groups	All locations	All	
<input type="checkbox"/> Name ↑	Type ↑	Resource Group ↑	Location ↑↓	Subscription ↑↓
<input type="checkbox"/> DXT-test-04-iso	<input type="checkbox"/> Virtual machine	[REDACTED]	eastus	[REDACTED]
<input checked="" type="checkbox"/> DemoVM-2	<input type="checkbox"/> Virtual machine	[REDACTED]	eastus	[REDACTED]
<input checked="" type="checkbox"/> DemoVM-3	<input type="checkbox"/> Virtual machine	[REDACTED]	southcentralus	[REDACTED]
<input type="checkbox"/> MCDemo1	<input type="checkbox"/> Virtual machine	[REDACTED]	eastus2	[REDACTED]
<input type="checkbox"/> myEUAPHost	<input type="checkbox"/> Dedicated host	[REDACTED]	centraluseuap	[REDACTED]
<input type="checkbox"/> my3rdHost	<input type="checkbox"/> Dedicated host	[REDACTED]	eastus2	[REDACTED]
<input type="checkbox"/> myWest2Host	<input type="checkbox"/> Dedicated host	[REDACTED]	westus2	[REDACTED]
<input type="checkbox"/> myHosteastUS	<input type="checkbox"/> Dedicated host	[REDACTED]	eastus2	[REDACTED]
<input type="checkbox"/> DDBLDLCLOUD02	<input type="checkbox"/> Virtual machine	[REDACTED]	westus	[REDACTED]

Ok  Cancel

## Check configuration

You can verify that the configuration was applied correctly or check to see any maintenance configuration that is currently assigned to a machine by going to the **Maintenance Configurations** and checking under the **Machines** tab. You should see any machine you have assigned the configuration in this tab.

**test | Machines** star ...

Maintenance Configuration

Search (Ctrl+ /) <> + Add machine X Remove machines ↻ Refresh

Overview Activity log Access control (IAM)

Tags

**Settings**

- Machines Schedule Updates Properties Locks
- Automation Tasks (preview) Export template

Manage resources assigned to this maintenance configuration. You can assign new resources, see the status of all resources currently assigned to the configuration, and by selecting resources below either apply pending maintenance or remove them from this configuration.

Name ↑↓	Type	Maintenance status ↑↓
<input type="checkbox"/> testvm13	<input type="checkbox"/> Virtual machine	Pending

## Check for pending updates

You can check if there are any updates pending for a maintenance configuration. In **Maintenance Configurations**, on the details for the configuration, click **Machines** and check **Maintenance status**.

**test | Machines** star ...

Maintenance Configuration

Search (Ctrl+ /) <> + Add machine X Remove machines ↻ Refresh

Overview Activity log Access control (IAM)

Tags

**Settings**

- Machines Schedule Updates Properties Locks
- Automation Tasks (preview) Export template

Manage resources assigned to this maintenance configuration. You can assign new resources, see the status of all resources currently assigned to the configuration, and by selecting resources below either apply pending maintenance or remove them from this configuration.

Name ↑↓	Type	Maintenance status ↑↓	Days until forced maintenance ↑↓	Resource group ↑↓	Location ↑↓	Subscription ↑↓
<input type="checkbox"/> testvm13	<input type="checkbox"/> Virtual machine	Pending	-	dean	Australia East	PLAYGROUND - IaaSExp - Team 01

## Delete a maintenance configuration

To delete a configuration, open the configuration details and click **Delete**.



test



...

Maintenance Configuration

Search (Ctrl+ /)

Delete

Overview

Activity log

Access control (IAM)

Tags

#### Settings

Machines

Schedule

Updates

Properties

Locks

#### Automation

Tasks (preview)

Export template

#### ^ Essentials

Resource group ([move](#)) :

Location : East US

Subscription ([move](#)) :

Subscription ID :

Tags ([edit](#)) : [Click here to add tags](#)



#### Manage machines

View and manage machines between this resource and other Azure resources such as dedicated hosts and virtual machines.

## Next steps

To learn more, see [Maintenance and updates](#).

# Azure Metadata Service: Scheduled Events for Linux VMs

9/21/2022 • 12 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets ✓ Uniform scale sets

Scheduled Events is an Azure Metadata Service that gives your application time to prepare for virtual machine (VM) maintenance. It provides information about upcoming maintenance events (for example, reboot) so that your application can prepare for them and limit disruption. It's available for all Azure Virtual Machines types, including PaaS and IaaS on both Windows and Linux.

For information about Scheduled Events on Windows, see [Scheduled Events for Windows VMs](#).

## NOTE

Scheduled Events is generally available in all Azure Regions. See [Version and Region Availability](#) for latest release information.

## Why use Scheduled Events?

Many applications can benefit from time to prepare for VM maintenance. The time can be used to perform application-specific tasks that improve availability, reliability, and serviceability, including:

- Checkpoint and restore.
- Connection draining.
- Primary replica failover.
- Removal from a load balancer pool.
- Event logging.
- Graceful shutdown.

With Scheduled Events, your application can discover when maintenance will occur and trigger tasks to limit its impact.

Scheduled Events provides events in the following use cases:

- [Platform initiated maintenance](#) (for example, VM reboot, live migration or memory preserving updates for host).
- Virtual machine is running on [degraded host hardware](#) that is predicted to fail soon.
- Virtual machine was running on a host that suffered a hardware failure.
- User-initiated maintenance (for example, a user restarts or redeploys a VM).
- [Spot VM](#) and [Spot scale set](#) instance evictions.

## The Basics

Metadata Service exposes information about running VMs by using a REST endpoint that's accessible from within the VM. The information is available via a nonroutable IP so that it's not exposed outside the VM.

### Scope

Scheduled events are delivered to:

- Standalone Virtual Machines.
- All the VMs in a cloud service.
- All the VMs in an availability set.
- All the VMs in an availability zone.
- All the VMs in a scale set placement group.

#### **NOTE**

Scheduled Events for all virtual machines (VMs) in a Fabric Controller (FC) tenant are delivered to all VMs in a FC tenant. FC tenant equates to a standalone VM, an entire Cloud Service, an entire Availability Set, and a Placement Group for a VM Scale Set (VMSS) regardless of Availability Zone usage. For example, if you have 100 VMs in a availability set and there is an update to one of them, the scheduled event will go to all 100, whereas if there are 100 single VMs in a zone, then event will only go to the VM which is getting impacted.

As a result, check the `Resources` field in the event to identify which VMs are affected.

#### **Endpoint discovery**

For VNET enabled VMs, Metadata Service is available from a static nonroutable IP, `169.254.169.254`. The full endpoint for the latest version of Scheduled Events is:

```
http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

If the VM is not created within a Virtual Network, the default cases for cloud services and classic VMs, additional logic is required to discover the IP address to use. To learn how to [discover the host endpoint](#), see this sample.

#### **Version and Region Availability**

The Scheduled Events service is versioned. Versions are mandatory; the current version is `2020-07-01`.

VERSION	RELEASE TYPE	REGIONS	RELEASE NOTES
2020-07-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Event Duration</li> </ul>
2019-08-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for EventSource</li> </ul>
2019-04-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Event Description</li> </ul>
2019-01-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for virtual machine scale sets EventType 'Terminate'</li> </ul>
2017-11-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Spot VM eviction EventType 'Preempt'</li> </ul>
2017-08-01	General Availability	All	<ul style="list-style-type: none"> <li>Removed prepended underscore from resource names for IaaS VMs</li> <li>Metadata header requirement enforced for all requests</li> </ul>

VERSION	RELEASE TYPE	REGIONS	RELEASE NOTES
2017-03-01	Preview	All	<ul style="list-style-type: none"> <li>Initial release</li> </ul>

#### NOTE

Previous preview releases of Scheduled Events supported {latest} as the api-version. This format is no longer supported and will be deprecated in the future.

## Enabling and Disabling Scheduled Events

Scheduled Events is enabled for your service the first time you make a request for events. You should expect a delayed response in your first call of up to two minutes.

Scheduled Events is disabled for your service if it does not make a request for 24 hours.

### User-initiated Maintenance

User-initiated VM maintenance via the Azure portal, API, CLI, or PowerShell results in a scheduled event. You then can test the maintenance preparation logic in your application, and your application can prepare for user-initiated maintenance.

If you restart a VM, an event with the type `Reboot` is scheduled. If you redeploy a VM, an event with the type `Redeploy` is scheduled. Typically events with a user event source can be immediately approved to avoid a delay on user-initiated actions.

## Use the API

### Headers

When you query Metadata Service, you must provide the header `Metadata:true` to ensure the request wasn't unintentionally redirected. The `Metadata:true` header is required for all scheduled events requests. Failure to include the header in the request results in a "Bad Request" response from Metadata Service.

### Query for events

You can query for scheduled events by making the following call:

#### Bash sample

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

#### Python sample

```
import json
import requests

metadata_url ="http://169.254.169.254/metadata/scheduledevents"
header = {'Metadata' : 'true'}
query_params = {'api-version':'2020-07-01'}

def get_scheduled_events():
    resp = requests.get(metadata_url, headers = header, params = query_params)
    data = resp.json()
    return data
```

A response contains an array of scheduled events. An empty array means that currently no events are scheduled. In the case where there are scheduled events, the response contains an array of events.

```
{
  "DocumentIncarnation": {IncarnationID},
  "Events": [
    {
      "EventId": {eventID},
      "EventType": "Reboot" | "Redeploy" | "Freeze" | "Preempt" | "Terminate",
      "ResourceType": "VirtualMachine",
      "Resources": [{resourceName}],
      "EventStatus": "Scheduled" | "Started",
      "NotBefore": {timeInUTC},
      "Description": {eventDescription},
      "EventSource" : "Platform" | "User",
      "DurationInSeconds" : {timeInSeconds},
    }
  ]
}
```

## Event Properties

PROPERTY	DESCRIPTION
Document Incarnation	<p>Integer that increases when the events array changes.</p> <p>Documents with the same incarnation contain the same event information, and the incarnation will be incremented when an event changes.</p>
EventId	<p>Globally unique identifier for this event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• 602d9444-d2cd-49c7-8624-8643e7171297</li> </ul>
EventType	<p>Impact this event causes.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>Freeze</code> : The Virtual Machine is scheduled to pause for a few seconds. CPU and network connectivity may be suspended, but there is no impact on memory or open files.</li> <li>• <code>Reboot</code> : The Virtual Machine is scheduled for reboot (non-persistent memory is lost). This event is made available on a best effort basis</li> <li>• <code>Redeploy</code> : The Virtual Machine is scheduled to move to another node (ephemeral disks are lost).</li> <li>• <code>Preempt</code> : The Spot Virtual Machine is being deleted (ephemeral disks are lost).</li> <li>• <code>Terminate</code> : The virtual machine is scheduled to be deleted.</li> </ul>
ResourceType	<p>Type of resource this event affects.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>VirtualMachine</code></li> </ul>
Resources	<p>List of resources this event affects.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• ["FrontEnd_IN_0", "BackEnd_IN_0"]</li> </ul>

PROPERTY	DESCRIPTION
EventStatus	<p>Status of this event.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>Scheduled</code> : This event is scheduled to start after the time specified in the <code>NotBefore</code> property.</li> <li>• <code>Started</code> : This event has started.</li> </ul> <p>No <code>Completed</code> or similar status is ever provided. The event is no longer returned when the event is finished.</p>
NotBefore	<p>Time after which this event can start. The event is guaranteed to not start before this time. Will be blank if the event has already started</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• Mon, 19 Sep 2016 18:29:47 GMT</li> </ul>
Description	<p>Description of this event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• Host server is undergoing maintenance.</li> </ul>
EventSource	<p>Initiator of the event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• <code>Platform</code> : This event is initiated by platform.</li> <li>• <code>User</code> : This event is initiated by user.</li> </ul>
DurationInSeconds	<p>The expected duration of the interruption caused by the event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• <code>9</code> : The interruption caused by the event will last for 9 seconds.</li> <li>• <code>-1</code> : The default value used if the impact duration is either unknown or not applicable.</li> </ul>

## Event Scheduling

Each event is scheduled a minimum amount of time in the future based on the event type. This time is reflected in an event's `NotBefore` property.

EVENT TYPE	MINIMUM NOTICE
Freeze	15 minutes
Reboot	15 minutes
Redeploy	10 minutes
Preempt	30 seconds

EVENT TYPE	MINIMUM NOTICE
Terminate	User Configurable: 5 to 15 minutes

#### NOTE

In some cases, Azure is able to predict host failure due to degraded hardware and will attempt to mitigate disruption to your service by scheduling a migration. Affected virtual machines will receive a scheduled event with a `NotBefore` that is typically a few days in the future. The actual time varies depending on the predicted failure risk assessment. Azure tries to give 7 days' advance notice when possible, but the actual time varies and might be smaller if the prediction is that there is a high chance of the hardware failing imminently. To minimize risk to your service in case the hardware fails before the system-initiated migration, we recommend that you self-redeploy your virtual machine as soon as possible.

#### NOTE

In the case the host node experiences a hardware failure Azure will bypass the minimum notice period and immediately begin the recovery process for affected virtual machines. This reduces recovery time in the case that the affected VMs are unable to respond. During the recovery process an event will be created for all impacted VMs with `EventType = Reboot` and `EventStatus = Started`.

## Polling frequency

You can poll the endpoint for updates as frequently or infrequently as you like. However, the longer the time between requests, the more time you potentially lose to react to an upcoming event. Most events have 5 to 15 minutes of advance notice, although in some cases advance notice might be as little as 30 seconds. To ensure that you have as much time as possible to take mitigating actions, we recommend that you poll the service once per second.

## Start an event

After you learn of an upcoming event and finish your logic for graceful shutdown, you can approve the outstanding event by making a `POST` call to Metadata Service with `EventId`. This call indicates to Azure that it can shorten the minimum notification time (when possible). The event may not start immediately upon approval, in some cases Azure will require the approval of all the VMs hosted on the node before proceeding with the event.

The following JSON sample is expected in the `POST` request body. The request should contain a list of `StartRequests`. Each `StartRequest` contains `EventId` for the event you want to expedite:

```
{
  "StartRequests" : [
    {
      "EventId": {EventId}
    }
  ]
}
```

The service will always return a 200 success code in the case of a valid event ID, even if it was already approved by a different VM. A 400 error code indicates that the request header or payload was malformed.

## Bash sample

```
curl -H Metadata:true -X POST -d '{"StartRequests": [{"EventId": "f020ba2e-3bc0-4c40-a10b-86575a9eabd5"}]}' http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

## Python sample

```
import json
import requests

def confirm_scheduled_event(event_id):
    # This payload confirms a single event with id event_id
    payload = json.dumps({"StartRequests": [{"EventId": event_id }]}))
    response = requests.post("http://169.254.169.254/metadata/scheduledevents",
                             headers = {'Metadata' : 'true'},
                             params = {'api-version':'2020-07-01'},
                             data = payload)
    return response.status_code
```

### NOTE

Acknowledging an event allows the event to proceed for all `Resources` in the event, not just the VM that acknowledges the event. Therefore, you can choose to elect a leader to coordinate the acknowledgement, which might be as simple as the first machine in the `Resources` field.

## Example Responses

The following is an example of a series of events that were seen by two VMs that were live migrated to another node.

The `DocumentIncarnation` is changing every time there is new information in `Events`. An approval of the event would allow the freeze to proceed for both WestNO\_0 and WestNO\_1. The `DurationInSeconds` of -1 indicates that the platform does not know how long the operation will take.

```
{
  "DocumentIncarnation": 1,
  "Events": [
    []
  }
}

{
  "DocumentIncarnation": 2,
  "Events": [
    {
      "EventId": "C7061BAC-AFDC-4513-B24B-AA5F13A16123",
      "EventStatus": "Scheduled",
      "EventType": "Freeze",
      "ResourceType": "VirtualMachine",
      "Resources": [
        "WestNO_0",
        "WestNO_1"
      ],
      "NotBefore": "Mon, 11 Apr 2022 22:26:58 GMT",
      "Description": "Virtual machine is being paused because of a memory-preserving Live Migration operation.",
      "EventSource": "Platform",
      "DurationInSeconds": -1
    }
  ]
}

{
  "DocumentIncarnation": 3,
  "Events": [
    {
      "EventId": "C7061BAC-AFDC-4513-B24B-AA5F13A16123",
      "EventStatus": "Started",
      "EventType": "Freeze",
      "ResourceType": "VirtualMachine",
      "Resources": [
        "WestNO_0",
        "WestNO_1"
      ],
      "NotBefore": "",
      "Description": "Virtual machine is being paused because of a memory-preserving Live Migration operation.",
      "EventSource": "Platform",
      "DurationInSeconds": -1
    }
  ]
}

{
  "DocumentIncarnation": 4,
  "Events": [
    []
  }
}
```

## Python Sample

The following sample queries Metadata Service for scheduled events and approves each outstanding event:

```
#!/usr/bin/python
import json
import requests
from time import sleep

# The URL to access the metadata service
```

```

metadata_url ="http://169.254.169.254/metadata/scheduledevents"
# This must be sent otherwise the request will be ignored
header = {'Metadata' : 'true'}
# Current version of the API
query_params = {'api-version':'2020-07-01'}

def get_scheduled_events():
    resp = requests.get(metadata_url, headers = header, params = query_params)
    data = resp.json()
    return data

def confirm_scheduled_event(event_id):
    # This payload confirms a single event with id event_id
    # You can confirm multiple events in a single request if needed
    payload = json.dumps({"StartRequests": [{"EventId": event_id }]})
    response = requests.post(metadata_url,
                            headers= header,
                            params = query_params,
                            data = payload)
    return response.status_code

def log(event):
    # This is an optional placeholder for logging events to your system
    print(event["Description"])
    return

def advanced_sample(last_document_incarnation):
    # Poll every second to see if there are new scheduled events to process
    # Since some events may have necessarily short warning periods, it is
    # recommended to poll frequently
    found_document_incarnation = last_document_incarnation
    while (last_document_incarnation == found_document_incarnation):
        sleep(1)
        payload = get_scheduled_events()
        found_document_incarnation = payload["DocumentIncarnation"]

        # We recommend processing all events in a document together,
        # even if you won't be actioning on them right away
        for event in payload["Events"]:

            # Events that have already started, logged for tracking
            if (event["EventStatus"] == "Started"):
                log(event)

            # Approve all user initiated events. These are typically created by an
            # administrator and approving them immediately can help to avoid delays
            # in admin actions
            elif (event["EventSource"] == "User"):
                confirm_scheduled_event(event["EventId"])

            # For this application, freeze events less than 9 seconds are considered
            # no impact. This will immediately approve them
            elif (event["EventType"] == "Freeze" and
                  int(event["DurationInSeconds"]) >= 0 and
                  int(event["DurationInSeconds"]) < 9):
                confirm_scheduled_event(event["EventId"])

            # Events that may be impactful (eg. Reboot or redeploy) may need custom
            # handling for your application
            else:
                #TODO Custom handling for impactful events
                log(event)
        print("Processed events from document: " + str(found_document_incarnation))
        return found_document_incarnation

def main():
    # This will track the last set of events seen
    last_document_incarnation = "-1"

```

```
input_text = "\n    Press 1 to poll for new events \n\n    Press 2 to exit \n\n"
program_exit = False

while program_exit == False:
    user_input = input(input_text)
    if (user_input == "1"):
        last_document_incarnation = advanced_sample(last_document_incarnation)
    elif (user_input == "2"):
        program_exit = True

if __name__ == '__main__':
    main()
```

## Next steps

- Review the Scheduled Events code samples in the [Azure Instance Metadata Scheduled Events GitHub repository](#).
- Review the Node.js Scheduled Events code samples in [Azure Samples GitHub repository](#).
- Read more about the APIs that are available in the [Instance Metadata Service](#).
- Learn about [planned maintenance for Linux virtual machines in Azure](#).
- Learn how to log scheduled events by using Azure Event Hubs in the [Azure Samples GitHub repository](#).

# Azure Metadata Service: Scheduled Events for Windows VMs

9/21/2022 • 12 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Scheduled Events is an Azure Metadata Service that gives your application time to prepare for virtual machine (VM) maintenance. It provides information about upcoming maintenance events (for example, reboot) so that your application can prepare for them and limit disruption. It's available for all Azure Virtual Machines types, including PaaS and IaaS on both Windows and Linux.

For information about Scheduled Events on Linux, see [Scheduled Events for Linux VMs](#).

## NOTE

Scheduled Events is generally available in all Azure Regions. See [Version and Region Availability](#) for latest release information.

## Why use Scheduled Events?

Many applications can benefit from time to prepare for VM maintenance. The time can be used to perform application-specific tasks that improve availability, reliability, and serviceability, including:

- Checkpoint and restore.
- Connection draining.
- Primary replica failover.
- Removal from a load balancer pool.
- Event logging.
- Graceful shutdown.

With Scheduled Events, your application can discover when maintenance will occur and trigger tasks to limit its impact.

Scheduled Events provides events in the following use cases:

- [Platform initiated maintenance](#) (for example, VM reboot, live migration or memory preserving updates for host).
- Virtual machine is running on [degraded host hardware](#) that is predicted to fail soon.
- Virtual machine was running on a host that suffered a hardware failure.
- User-initiated maintenance (for example, a user restarts or redeploys a VM).
- [Spot VM](#) and [Spot scale set](#) instance evictions.

## The Basics

Metadata Service exposes information about running VMs by using a REST endpoint that's accessible from within the VM. The information is available via a nonroutable IP so that it's not exposed outside the VM.

### Scope

Scheduled events are delivered to:

- Standalone Virtual Machines.
- All the VMs in a cloud service.
- All the VMs in an availability set.
- All the VMs in an availability zone.
- All the VMs in a scale set placement group.

#### **NOTE**

Scheduled Events for all virtual machines (VMs) in a Fabric Controller (FC) tenant are delivered to all VMs in a FC tenant. FC tenant equates to a standalone VM, an entire Cloud Service, an entire Availability Set, and a Placement Group for a VM Scale Set (VMSS) regardless of Availability Zone usage.

As a result, check the `Resources` field in the event to identify which VMs are affected.

#### **Endpoint discovery**

For VNET enabled VMs, Metadata Service is available from a static nonroutable IP, `169.254.169.254`. The full endpoint for the latest version of Scheduled Events is:

```
http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

If the VM is not created within a Virtual Network, the default cases for cloud services and classic VMs, additional logic is required to discover the IP address to use. To learn how to [discover the host endpoint](#), see this sample.

#### **Version and region availability**

The Scheduled Events service is versioned. Versions are mandatory; the current version is `2020-07-01`.

VERSION	RELEASE TYPE	REGIONS	RELEASE NOTES
2020-07-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Event Duration</li> </ul>
2019-08-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for EventSource</li> </ul>
2019-04-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Event Description</li> </ul>
2019-01-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for virtual machine scale sets EventType 'Terminate'</li> </ul>
2017-11-01	General Availability	All	<ul style="list-style-type: none"> <li>Added support for Spot VM eviction EventType 'Preempt'</li> </ul>
2017-08-01	General Availability	All	<ul style="list-style-type: none"> <li>Removed prepended underscore from resource names for IaaS VMs</li> <li>Metadata header requirement enforced for all requests</li> </ul>
2017-03-01	Preview	All	<ul style="list-style-type: none"> <li>Initial release</li> </ul>

#### **NOTE**

Previous preview releases of Scheduled Events supported {latest} as the api-version. This format is no longer supported and will be deprecated in the future.

## **Enabling and disabling Scheduled Events**

Scheduled Events is enabled for your service the first time you make a request for events. You should expect a delayed response in your first call of up to two minutes.

Scheduled Events is disabled for your service if it does not make a request for 24 hours.

### **User-initiated maintenance**

User-initiated VM maintenance via the Azure portal, API, CLI, or PowerShell results in a scheduled event. You then can test the maintenance preparation logic in your application, and your application can prepare for user-initiated maintenance.

If you restart a VM, an event with the type `Reboot` is scheduled. If you redeploy a VM, an event with the type `Redeploy` is scheduled. Typically events with a user event source can be immediately approved to avoid a delay on user-initiated actions.

## **Use the API**

### **Headers**

When you query Metadata Service, you must provide the header `Metadata:true` to ensure the request wasn't unintentionally redirected. The `Metadata:true` header is required for all scheduled events requests. Failure to include the header in the request results in a "Bad Request" response from Metadata Service.

### **Query for events**

You can query for scheduled events by making the following call:

#### **Bash sample**

```
curl -H Metadata:true http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

#### **PowerShell sample**

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -Uri "http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01" | ConvertTo-Json -Depth 64
```

#### **Python sample**

```
import json
import requests

metadata_url = "http://169.254.169.254/metadata/scheduledevents"
header = {'Metadata' : 'true'}
query_params = {'api-version':'2020-07-01'}

def get_scheduled_events():
    resp = requests.get(metadata_url, headers = header, params = query_params)
    data = resp.json()
    return data
```

A response contains an array of scheduled events. An empty array means that currently no events are scheduled. In the case where there are scheduled events, the response contains an array of events.

```
{
  "DocumentIncarnation": {IncarnationID},
  "Events": [
    {
      "EventId": {eventID},
      "EventType": "Reboot" | "Redeploy" | "Freeze" | "Preempt" | "Terminate",
      "ResourceType": "VirtualMachine",
      "Resources": [{resourceName}],
      "EventStatus": "Scheduled" | "Started",
      "NotBefore": {timeInUTC},
      "Description": {eventDescription},
      "EventSource" : "Platform" | "User",
      "DurationInSeconds" : {timeInSeconds},
    }
  ]
}
```

## Event properties

PROPERTY	DESCRIPTION
Document Incarnation	<p>Integer that increases when the events array changes.</p> <p>Documents with the same incarnation contain the same event information, and the incarnation will be incremented when an event changes.</p>
EventId	<p>Globally unique identifier for this event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• 602d9444-d2cd-49c7-8624-8643e7171297</li> </ul>
EventType	<p>Impact this event causes.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>Freeze</code> : The Virtual Machine is scheduled to pause for a few seconds. CPU and network connectivity may be suspended, but there is no impact on memory or open files.</li> <li>• <code>Reboot</code> : The Virtual Machine is scheduled for reboot (non-persistent memory is lost).</li> <li>• <code>Redeploy</code> : The Virtual Machine is scheduled to move to another node (ephemeral disks are lost).</li> <li>• <code>Preempt</code> : The Spot Virtual Machine is being deleted (ephemeral disks are lost). This event is made available on a best effort basis</li> <li>• <code>Terminate</code> : The virtual machine is scheduled to be deleted.</li> </ul>
ResourceType	<p>Type of resource this event affects.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>VirtualMachine</code></li> </ul>
Resources	<p>List of resources this event affects.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• ["FrontEnd_IN_0", "BackEnd_IN_0"]</li> </ul>

PROPERTY	DESCRIPTION
EventStatus	<p>Status of this event.</p> <p>Values:</p> <ul style="list-style-type: none"> <li>• <code>Scheduled</code> : This event is scheduled to start after the time specified in the <code>NotBefore</code> property.</li> <li>• <code>Started</code> : This event has started.</li> </ul> <p>No <code>Completed</code> or similar status is ever provided. The event is no longer returned when the event is finished.</p>
NotBefore	<p>Time after which this event can start. The event is guaranteed to not start before this time. Will be blank if the event has already started</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• Mon, 19 Sep 2016 18:29:47 GMT</li> </ul>
Description	<p>Description of this event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• Host server is undergoing maintenance.</li> </ul>
EventSource	<p>Initiator of the event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• <code>Platform</code> : This event is initiated by platform.</li> <li>• <code>User</code> : This event is initiated by user.</li> </ul>
DurationInSeconds	<p>The expected duration of the interruption caused by the event.</p> <p>Example:</p> <ul style="list-style-type: none"> <li>• <code>9</code> : The interruption caused by the event will last for 9 seconds.</li> <li>• <code>-1</code> : The default value used if the impact duration is either unknown or not applicable.</li> </ul>

## Event scheduling

Each event is scheduled a minimum amount of time in the future based on the event type. This time is reflected in an event's `NotBefore` property.

EVENT TYPE	MINIMUM NOTICE
Freeze	15 minutes
Reboot	15 minutes
Redeploy	10 minutes
Preempt	30 seconds

EVENT TYPE	MINIMUM NOTICE
Terminate	User Configurable: 5 to 15 minutes

#### NOTE

In some cases, Azure is able to predict host failure due to degraded hardware and will attempt to mitigate disruption to your service by scheduling a migration. Affected virtual machines will receive a scheduled event with a `NotBefore` that is typically a few days in the future. The actual time varies depending on the predicted failure risk assessment. Azure tries to give 7 days' advance notice when possible, but the actual time varies and might be smaller if the prediction is that there is a high chance of the hardware failing imminently. To minimize risk to your service in case the hardware fails before the system-initiated migration, we recommend that you self-redeploy your virtual machine as soon as possible.

#### NOTE

In the case the host node experiences a hardware failure Azure will bypass the minimum notice period and immediately begin the recovery process for affected virtual machines. This reduces recovery time in the case that the affected VMs are unable to respond. During the recovery process an event will be created for all impacted VMs with `EventType = Reboot` and `EventStatus = Started`.

## Polling frequency

You can poll the endpoint for updates as frequently or infrequently as you like. However, the longer the time between requests, the more time you potentially lose to react to an upcoming event. Most events have 5 to 15 minutes of advance notice, although in some cases advance notice might be as little as 30 seconds. To ensure that you have as much time as possible to take mitigating actions, we recommend that you poll the service once per second.

## Start an event

After you learn of an upcoming event and finish your logic for graceful shutdown, you can approve the outstanding event by making a `POST` call to Metadata Service with `EventId`. This call indicates to Azure that it can shorten the minimum notification time (when possible). The event may not start immediately upon approval, in some cases Azure will require the approval of all the VMs hosted on the node before proceeding with the event.

The following JSON sample is expected in the `POST` request body. The request should contain a list of `StartRequests`. Each `StartRequest` contains `EventId` for the event you want to expedite:

```
{
  "StartRequests" : [
    {
      "EventId": {EventId}
    }
  ]
}
```

The service will always return a 200 success code in the case of a valid event ID, even if it was already approved by a different VM. A 400 error code indicates that the request header or payload was malformed.

## Bash sample

```
curl -H Metadata:true -X POST -d '{"StartRequests": [{"EventId": "f020ba2e-3bc0-4c40-a10b-86575a9eabd5"}]}' http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01
```

## PowerShell sample

```
Invoke-RestMethod -Headers @{"Metadata" = "true"} -Method POST -body '{"StartRequests": [{"EventId": "5DD55B64-45AD-49D3-BBC9-F57D4EA97BD7"}]}' -Uri http://169.254.169.254/metadata/scheduledevents?api-version=2020-07-01 | ConvertTo-Json -Depth 64
```

## Python sample

```
import json
import requests

def confirm_scheduled_event(event_id):
    # This payload confirms a single event with id event_id
    payload = json.dumps({"StartRequests": [{"EventId": event_id}]})
    response = requests.post("http://169.254.169.254/metadata/scheduledevents",
                             headers = {'Metadata' : 'true'},
                             params = {'api-version':'2020-07-01'},
                             data = payload)
    return response.status_code
```

### NOTE

Acknowledging an event allows the event to proceed for all `Resources` in the event, not just the VM that acknowledges the event. Therefore, you can choose to elect a leader to coordinate the acknowledgement, which might be as simple as the first machine in the `Resources` field.

## Example responses

The following is an example of a series of events that were seen by two VMs that were live migrated to another node.

The `DocumentIncarnation` is changing every time there is new information in `Events`. An approval of the event would allow the freeze to proceed for both WestNO\_0 and WestNO\_1. The `DurationInSeconds` of -1 indicates that the platform does not know how long the operation will take.

```
{
  "DocumentIncarnation": 1,
  "Events": [
    []
  }
}

{
  "DocumentIncarnation": 2,
  "Events": [
    {
      "EventId": "C7061BAC-AFDC-4513-B24B-AA5F13A16123",
      "EventStatus": "Scheduled",
      "EventType": "Freeze",
      "ResourceType": "VirtualMachine",
      "Resources": [
        "WestNO_0",
        "WestNO_1"
      ],
      "NotBefore": "Mon, 11 Apr 2022 22:26:58 GMT",
      "Description": "Virtual machine is being paused because of a memory-preserving Live Migration operation.",
      "EventSource": "Platform",
      "DurationInSeconds": -1
    }
  ]
}

{
  "DocumentIncarnation": 3,
  "Events": [
    {
      "EventId": "C7061BAC-AFDC-4513-B24B-AA5F13A16123",
      "EventStatus": "Started",
      "EventType": "Freeze",
      "ResourceType": "VirtualMachine",
      "Resources": [
        "WestNO_0",
        "WestNO_1"
      ],
      "NotBefore": "",
      "Description": "Virtual machine is being paused because of a memory-preserving Live Migration operation.",
      "EventSource": "Platform",
      "DurationInSeconds": -1
    }
  ]
}

{
  "DocumentIncarnation": 4,
  "Events": [
    []
  }
}
```

## Python Sample

The following sample queries Metadata Service for scheduled events and approves each outstanding event:

```
#!/usr/bin/python
import json
import requests
from time import sleep

# The URL to access the metadata service
```

```

metadata_url ="http://169.254.169.254/metadata/scheduledevents"
# This must be sent otherwise the request will be ignored
header = {'Metadata' : 'true'}
# Current version of the API
query_params = {'api-version':'2020-07-01'}

def get_scheduled_events():
    resp = requests.get(metadata_url, headers = header, params = query_params)
    data = resp.json()
    return data

def confirm_scheduled_event(event_id):
    # This payload confirms a single event with id event_id
    # You can confirm multiple events in a single request if needed
    payload = json.dumps({"StartRequests": [{"EventId": event_id }]})
    response = requests.post(metadata_url,
                            headers= header,
                            params = query_params,
                            data = payload)
    return response.status_code

def log(event):
    # This is an optional placeholder for logging events to your system
    print(event["Description"])
    return

def advanced_sample(last_document_incarnation):
    # Poll every second to see if there are new scheduled events to process
    # Since some events may have necessarily short warning periods, it is
    # recommended to poll frequently
    found_document_incarnation = last_document_incarnation
    while (last_document_incarnation == found_document_incarnation):
        sleep(1)
        payload = get_scheduled_events()
        found_document_incarnation = payload["DocumentIncarnation"]

        # We recommend processing all events in a document together,
        # even if you won't be actioning on them right away
        for event in payload["Events"]:

            # Events that have already started, logged for tracking
            if (event["EventStatus"] == "Started"):
                log(event)

            # Approve all user initiated events. These are typically created by an
            # administrator and approving them immediately can help to avoid delays
            # in admin actions
            elif (event["EventSource"] == "User"):
                confirm_scheduled_event(event["EventId"])

            # For this application, freeze events less than 9 seconds are considered
            # no impact. This will immediately approve them
            elif (event["EventType"] == "Freeze" and
                  int(event["DurationInSeconds"]) >= 0 and
                  int(event["DurationInSeconds"]) < 9):
                confirm_scheduled_event(event["EventId"])

            # Events that may be impactful (eg. Reboot or redeploy) may need custom
            # handling for your application
            else:
                #TODO Custom handling for impactful events
                log(event)
        print("Processed events from document: " + str(found_document_incarnation))
        return found_document_incarnation

def main():
    # This will track the last set of events seen
    last_document_incarnation = "-1"

```

```
input_text = "\n    Press 1 to poll for new events \n\n    Press 2 to exit \n\n"
program_exit = False

while program_exit == False:
    user_input = input(input_text)
    if (user_input == "1"):
        last_document_incarnation = advanced_sample(last_document_incarnation)
    elif (user_input == "2"):
        program_exit = True

if __name__ == '__main__':
    main()
```

## Next steps

- Review the Scheduled Events code samples in the [Azure Instance Metadata Scheduled Events GitHub repository](#).
- Review the Node.js Scheduled Events code samples in [Azure Samples GitHub repository](#).
- Read more about the APIs that are available in the [Instance Metadata Service](#).
- Learn about [planned maintenance for Windows virtual machines in Azure](#).
- Learn how to [monitor scheduled events for your VMs through Log Analytics](#).
- Learn how to log scheduled events using Azure Event Hub in the [Azure Samples GitHub repository](#).

# Monitor scheduled events for your Azure VMs

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

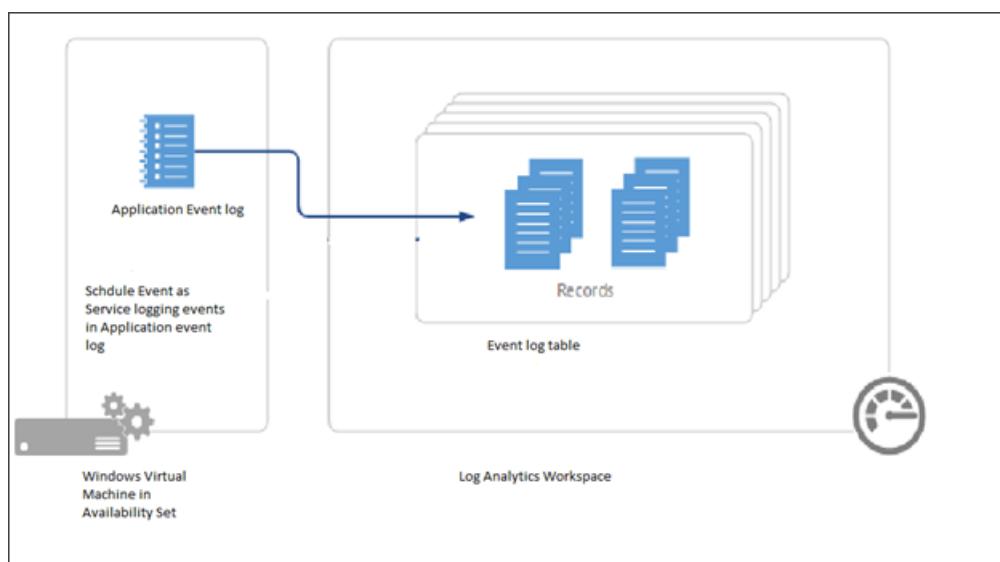
Updates are applied to different parts of Azure every day, to keep the services running on them secure, and up-to-date. In addition to planned updates, unplanned events may also occur. For example, if any hardware degradation or fault is detected, Azure services may need to perform unplanned maintenance. Using live migration, memory preserving updates and generally keeping a strict bar on the impact of updates, in most cases these events are almost transparent to customers, and they have no impact or at most cause a few seconds of virtual machine freeze. However, for some applications, even a few seconds of virtual machine freeze could cause an impact. Knowing in advance about upcoming Azure maintenance is important, to ensure the best experience for those applications. [Scheduled Events service](#) provides you a programmatic interface to be notified about upcoming maintenance, and enables you to gracefully handle the maintenance.

In this article, we will show how you can use scheduled events to be notified about maintenance events that could be affecting your VMs and build some basic automation that can help with monitoring and analysis.

## Routing scheduled events to Log Analytics

Scheduled Events is available as part of the [Azure Instance Metadata Service](#), which is available on every Azure virtual machine. Customers can write automation to query the endpoint of their virtual machines to find scheduled maintenance notifications and perform mitigations, like saving the state and taking the virtual machine out of rotation. We recommend building automation to record the Scheduled Events so you can have an auditing log of Azure maintenance events.

In this article, we will walk you through how to capture maintenance Scheduled Events to Log Analytics. Then, we will trigger some basic notification actions, like sending an email to your team and getting a historical view of all events that have affected your virtual machines. For the event aggregation and automation we will use [Log Analytics](#), but you can use any monitoring solution to collect these logs and trigger automation.



## Prerequisites

For this example, you will need to create a [Windows Virtual Machine in an Availability Set](#). Scheduled Events provide notifications about changes that can affect any of the virtual machines in your availability set, Cloud

Service, Virtual Machine Scale Set or standalone VMs. We will be running a [service](#) that polls for scheduled events on one of the VMs that will act as a collector, to get events for all of the other VMs in the availability set.

Don't delete the group resource group at the end of the tutorial.

You will also need to [create a Log Analytics workspace](#) that we will use to aggregate information from the VMs in the availability set.

## Set up the environment

You should now have 2 initial VMs in an availability set. Now we need to create a 3rd VM, called `myCollectorVM`, in the same availability set.

```
New-AzVm ` 
  -ResourceGroupName "myResourceGroupAvailability" ` 
  -Name "myCollectorVM" ` 
  -Location "East US" ` 
  -VirtualNetworkName "myVnet" ` 
  -SubnetName "mySubnet" ` 
  -SecurityGroupName "myNetworkSecurityGroup" ` 
  -OpenPorts 3389 ` 
  -PublicIpAddressName "myPublicIpAddress3" ` 
  -AvailabilitySetName "myAvailabilitySet" ` 
  -Credential $cred
```

Download the installation .zip file of the project from [GitHub](#).

Connect to `myCollectorVM` and copy the .zip file to the virtual machine and extract all of the files. On your VM, open a PowerShell prompt. Move your prompt into the folder containing `SchService.ps1`, for example:

```
PS C:\Users\azureuser\AzureScheduledEventsService-master\AzureScheduledEventsService-master\Powershell>, and
set up the service.
```

```
.\SchService.ps1 -Setup
```

Start the service.

```
.\SchService.ps1 -Start
```

The service will now start polling every 10 seconds for any scheduled events and approve the events to expedite the maintenance. Freeze, Reboot, Redeploy, and Preempt are the events captured by Schedule events. Note that you can extend the script to trigger some mitigations prior to approving the event.

Validate the service status and make sure it is running.

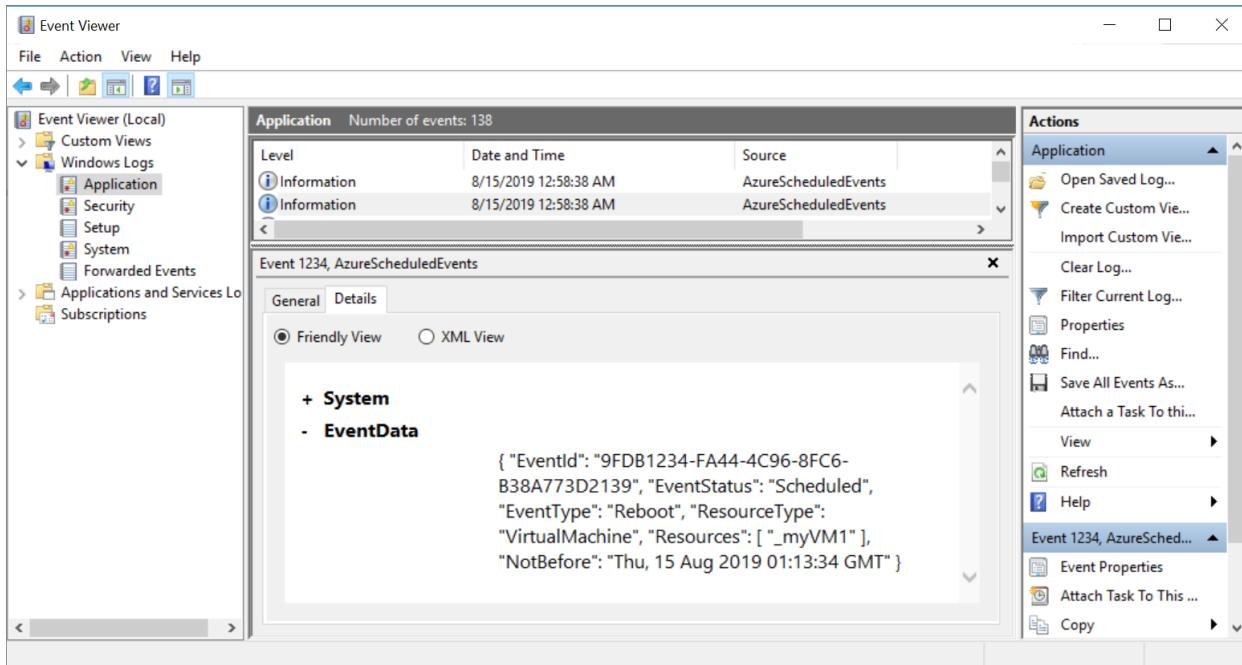
```
.\SchService.ps1 -status
```

This should return `Running`.

The service will now start polling every 10 seconds for any scheduled events and approve the events to expedite the maintenance. Freeze, Reboot, Redeploy and Preempt are the events captured by Schedule events. You can extend the script to trigger some mitigations prior to approving the event.

When any of the above events are captured by Schedule Event service, it will get logged in the Application Event Log Event Status, Event Type, Resources (Virtual machine names) and NotBefore (minimum notice period). You can locate the events with ID 1234 in the Application Event Log.

Once the service is set up and started, it will log events in the Windows Application logs. To verify this works, restart one of the virtual machines in the availability set and you should see an event being logged in Event viewer in Windows Logs > Application log showing the VM restarted.



When events are captured by the Schedule Event service, it will get logged in the application even log with Event Status, Event Type, Resources (VM name) and NotBefore (minimum notice period). You can locate the events with ID 1234 in the Application Event Log.

#### NOTE

In this example, the virtual machines were are in an availability set, which enabled us to designate a single virtual machine as the collector to listen and route scheduled events to our log analytics works space. If you have standalone virtual machines, you can run the service on every virtual machine, and then connect them individually to your log analytics workspace.

For our set up, we chose Windows, but you can design a similar solution on Linux.

At any point you can stop/remove the Scheduled Event Service by using the switches `-stop` and `-remove`.

## Connect to the workspace

We now want to connect a Log Analytics Workspace to the collector VM. The Log Analytics workspace acts as a repository and we will configure event log collection to capture the application logs from the collector VM.

To route the Scheduled Events to the Events Log, which will be saved as Application log by our service, you will need to connect your virtual machine to your Log Analytics workspace.

1. Open the page for the workspace you created.
2. Under Connect to a data source select **Azure virtual machines (VMs)**.

## Get started with Log Analytics

Log Analytics collects data from a variety of sources and uses a powerful query language to give you insights into the operation of your applications and resources. Use Azure Monitor to access the complete set of tools for monitoring all of your Azure resources.

### 1 Connect a data source

Select one or more data sources to connect to the workspace

[Azure virtual machines \(VMs\)](#)

[Windows, Linux and other sources](#)

[Azure Activity logs](#)

3. Search for and select **myCollectorVM**.

4. On the new page for **myCollectorVM**, select **Connect**.

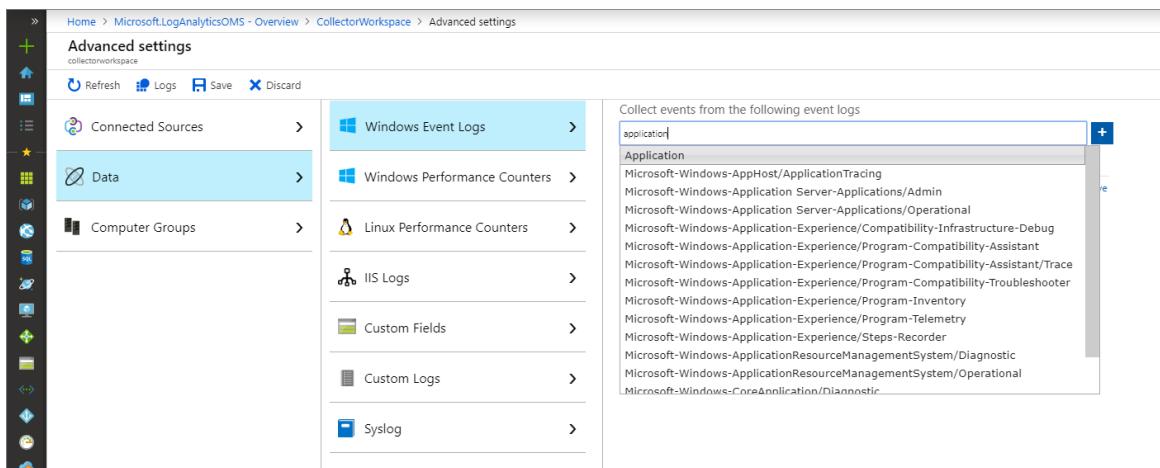
This will install the [Microsoft Monitoring agent](#) in your virtual machine. It will take a few minutes to connect your VM to the workspace and install the extension.

## Configure the workspace

1. Open the page for your workspace and select **Advanced settings**.

2. Select **Data** from the left menu, then select **Windows Event Logs**.

3. In **Collect from the following event logs**, start typing *application* and then select **Application** from the list.



4. Leave **ERROR**, **WARNING**, and **INFORMATION** selected and then select **Save** to save the settings.

### NOTE

There will be some delay, and it may take up to 10 minutes before the log is available.

## Creating an alert rule with Azure Monitor

Once the events are pushed to Log Analytics, you can run the following [query](#) to look for the schedule Events.

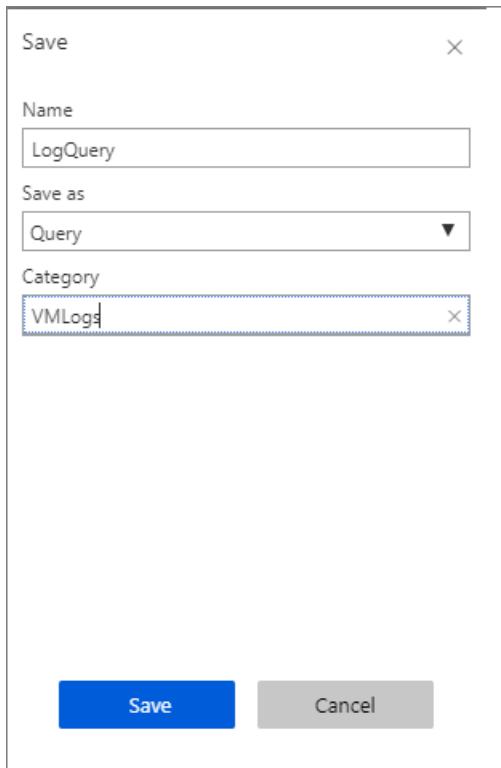
1. At the top of the page, select **Logs** and paste the following into the text box:

```

Event
| where EventLog == "Application" and Source contains "AzureScheduledEvents" and RenderedDescription
contains "Scheduled" and RenderedDescription contains "EventStatus"
| project TimeGenerated, RenderedDescription
| extend ReqJson= parse_json(RenderedDescription)
| extend EventId = ReqJson["EventId"]
,EventStatus = ReqJson["EventStatus"]
,EventType = ReqJson["EventType"]
,NotBefore = ReqJson["NotBefore"]
,ResourceType = ReqJson["ResourceType"]
,Resources = ReqJson["Resources"]
| project-away RenderedDescription,ReqJson

```

2. Select **Save**, and then type `LogQuery` for the name, leave **Query** as the type, type `VMLogs` as the **Category**, and then select **Save**.



3. Select **New alert rule**.
4. In the **Create rule** page, leave `collectorworkspace` as the **Resource**.
5. Under **Condition**, select the entry *Whenever the customer log search is <login undefined>*. The **Configure signal logic** page will open.
6. Under **Threshold value**, enter *0* and then select **Done**.
7. Under **Actions**, select **Create action group**. The **Add action group** page will open.
8. In **Action group name**, type *myActionGroup*.
9. In **Short name**, type *myActionGroup*.
10. In **Resource group**, select **myResourceGroupAvailability**.
11. Under **Actions**, in **ACTION NAME** type **Email**, and then select **Email/SMS/Push/Voice**. The **Email/SMS/Push/Voice** page will open.
12. Select **Email**, type in your e-mail address, then select **OK**.
13. In the **Add action group** page, select **OK**.

14. In the **Create rule** page, under **ALERT DETAILS**, type *myAlert* for the **Alert rule name**, and then type *Email alert rule* for the **Description**.
15. When you are finished, select **Create alert rule**.
16. Restart one of the VMs in the availability set. Within a few minutes, you should get an e-mail that the alert has been triggered.

To manage your alert rules, go to the resource group, select **Alerts** from the left menu, and then select **Manage alert rules** from the top of the page.

## Next steps

To learn more, see the [Scheduled events service](#) page on GitHub.

# Tutorial: Monitor changes and update a Linux virtual machine in Azure

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

Azure [Change Tracking](#) allows you to easily identify changes and [Update Management](#) allows you to manage operating system updates for your Azure Linux VMs.

In this tutorial, you learn how to:

- Manage Linux updates
- Monitor changes and inventory

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This tutorial requires version 2.0.30 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create VM

To see diagnostics and metrics in action, you need a VM. First, create a resource group with `az group create`. The following example creates a resource group named `myResourceGroupMonitor` in the `eastus` location.

```
az group create --name myResourceGroupMonitor --location eastus
```

Now create a VM with `az vm create`. The following example creates a VM named `myVM` and generates SSH keys if they do not already exist in `~/.ssh/`.

```
az vm create \
--resource-group myResourceGroupMonitor \
--name myVM \
--image UbuntuLTS \
--admin-username azureuser \
--generate-ssh-keys
```

## Manage software updates

Update management allows you to manage updates and patches for your Azure Linux VMs. Directly from your VM, you can quickly assess the status of available updates, schedule installation of required updates, and review deployment results to verify updates were applied successfully to the VM.

For pricing information, see [Automation pricing for Update management](#)

### Enable Update management

Enable Update management for your VM:

1. On the left-hand side of the screen, select **Virtual machines**.
2. From the list, select a VM.
3. On the VM screen, in the **Operations** section, select **Update management**. The **Enable Update Management** screen opens.

Validation is performed to determine if Update management is enabled for this VM. The validation includes checks for a Log Analytics workspace and linked Automation account, and if the solution is in the workspace.

A [Log Analytics](#) workspace is used to collect data that is generated by features and services such as Update management. The workspace provides a single location to review and analyze data from multiple sources. To perform additional actions on VMs that require updates, Azure Automation allows you to run runbooks against VMs, such as download and apply updates.

The validation process also checks to see if the VM is provisioned with the Log Analytics agent and Automation hybrid runbook worker. This agent is used to communicate with the VM and obtain information about the update status.

Choose the Log Analytics workspace and automation account and select **Enable** to enable the solution. The solution takes up to 15 minutes to enable.

If any of the following prerequisites were found to be missing during onboarding, they're automatically added:

- [Log Analytics workspace](#)
- [Automation account](#)
- A [Hybrid runbook worker](#) is enabled on the VM

The **Update Management** screen opens. Configure the location, Log Analytics workspace and Automation account to use and select **Enable**. If the fields are grayed out, that means another automation solution is enabled for the VM and the same workspace and Automation account must be used.

Enabling the solution can take up to 15 minutes. During this time, you shouldn't close the browser window. After the solution is enabled, information about missing updates on the VM flows to Azure Monitor logs. It can take between 30 minutes and 6 hours for the data to be available for analysis.

## View update assessment

After **Update management** is enabled, the **Update management** screen appears. After the evaluation of updates is complete, you see a list of missing updates on the **Missing updates** tab.

UPDATE NAME	CLASSIFICATION	INFORMATION LINK
apport	critical	<a href="#">View details</a>
sudo	critical	<a href="#">View details</a>
linux-firmware	critical	<a href="#">View details</a>
dosfstools	security	<a href="#">View details</a>
bash	security	<a href="#">View details</a>
cpio	security	<a href="#">View details</a>
curl	security	<a href="#">View details</a>
patch	security	<a href="#">View details</a>

## Schedule an update deployment

To install updates, schedule a deployment that follows your release schedule and service window. You can choose which update types to include in the deployment. For example, you can include critical or security updates and exclude update rollups.

Schedule a new Update Deployment for the VM by clicking **Schedule update deployment** at the top of the **Update management** screen. In the **New update deployment** screen, specify the following information:

To create a new update deployment, select **Schedule update deployment**. The **New update deployment**

page opens. Enter values for the properties described in the following table and then click **Create**:

PROPERTY	DESCRIPTION
Name	Unique name to identify the update deployment.
Operating System	Linux or Windows
Groups to update	<p>For Azure machines, define a query based on a combination of subscription, resource groups, locations, and tags to build a dynamic group of Azure VMs to include in your deployment.</p> <p>For Non-Azure machines, select an existing saved search to select a group of Non-Azure machines to include in the deployment.</p> <p>To learn more, see <a href="#">Dynamic Groups</a></p>
Machines to update	<p>Select a Saved search, Imported group, or pick Machine from the drop-down and select individual machines. If you choose <b>Machines</b>, the readiness of the machine is shown in the <b>UPDATE AGENT READINESS</b> column.</p> <p>To learn about the different methods of creating computer groups in Azure Monitor logs, see <a href="#">Computer groups in Azure Monitor logs</a></p>
Update classifications	Select all the update classifications that you need
Include/exclude updates	This opens the <b>Include/Exclude</b> page. Updates to be included or excluded are on separate tabs. For more information on how inclusion is handled, see <a href="#">Schedule an Update Deployment</a>
Schedule settings	Select the time to start, and select either Once or recurring for the recurrence
Pre-scripts + Post-scripts	Select the scripts to run before and after your deployment
Maintenance window	Number of minutes set for updates. The value can't be less than 30 minutes and no more than 6 hours
Reboot control	<p>Determines how reboots should be handled. Available options are:</p> <ul style="list-style-type: none"> <li>Reboot if required (Default)</li> <li>Always reboot</li> <li>Never reboot</li> <li>Only reboot - will not install updates</li> </ul>

Update Deployments can also be created programmatically. To learn how to create an Update Deployment with the REST API, see [Software Update Configurations - Create](#). There is also a sample runbook that can be used to create a weekly Update Deployment. To learn more about this runbook, see [Create a weekly update deployment for one or more VMs in a resource group](#).

After you have completed configuring the schedule, click **Create** button and you return to the status dashboard. Notice that the **Scheduled** table shows the deployment schedule you created.

#### **View results of an update deployment**

After the scheduled deployment starts, you can see the status for that deployment on the **Update deployments** tab on the **Update management** screen. If it is currently running, its status shows as **In progress**. After it completes, if successful, it changes to **Succeeded**. If there is a failure with one or more updates in the deployment, the status is **Partially failed**. Select the completed update deployment to see the dashboard for that update deployment.

The screenshot shows the Update management dashboard for a VM named Marketing10. The deployment run has completed successfully. The deployment results summary indicates 89 Updates, all of which were Succeeded. The detailed updates status table lists five specific updates, each with a green checkmark indicating success. Below the table, there are navigation links for page numbers 1 through 18. The bottom section of the dashboard features three tabs: All Logs, Output, and Errors. The Errors tab is selected, showing 0 errors.

UPDATE NAME	STATUS
NetworkManager-libnm.x86_64	Succeeded
NetworkManager-tui.x86_64	Succeeded
NetworkManager-team.x86_64	Succeeded
NetworkManager.x86_64	Succeeded
WALinuxAgent.noarch	Succeeded

In **Update results** tile is a summary of the total number of updates and deployment results on the VM. In the table to the right is a detailed breakdown of each update and the installation results, which could be one of the following values:

- **Not attempted** - the update was not installed because there was insufficient time available based on the maintenance window duration defined.
- **Succeeded** - the update succeeded
- **Failed** - the update failed

Select **All logs** to see all log entries that the deployment created.

Select the **Output** tile to see job stream of the runbook responsible for managing the update deployment on the target VM.

Select **Errors** to see detailed information about any errors from the deployment.

## Monitor changes and inventory

You can collect and view inventory for software, files, Linux daemons, Windows Services, and Windows registry keys on your computers. Tracking the configurations of your machines can help you pinpoint operational issues across your environment and better understand the state of your machines.

### Enable Change and Inventory management

Enable Change and Inventory management for your VM:

1. On the left-hand side of the screen, select **Virtual machines**.
2. From the list, select a VM.
3. On the VM screen, in the **Operations** section, select **Inventory or Change tracking**. The **Enable Change Tracking and Inventory** screen opens.

Configure the location, Log Analytics workspace and Automation account to use and select **Enable**. If the fields are grayed out, that means another automation solution is enabled for the VM and the same workspace and Automation account must be used. Even though the solutions are separate on the menu, they are the same.

solution. Enabling one enables both for your VM.

The screenshot shows the Azure portal interface for a virtual machine named 'myVM'. The left sidebar has a search bar and several navigation items: Auto-shutdown, Backup, Disaster recovery (Preview), Update management, Inventory (which is selected and highlighted in blue), and Change tracking. Below these are sections for MONITORING: Metrics, Alert rules, Diagnostics settings, Advisor recommendations, and Diagram. The main content area is titled 'Inventory' and contains the following text:  
Enable consistent control and compliance of this VM with Change Tracking and Inventory.  
This service is included with Azure virtual machines. You only pay for logs stored in Log Analytics.  
This service requires a Log Analytics workspace and an Automation account. You can use your existing workspace and account or let us configure the nearest workspace and account for use.  
Below this are three dropdown menus: Location (set to East US), Log Analytics workspace (set to defaultworkspace), and Automation account (set to Automate). A large blue 'Enable' button is at the bottom.

After the solution has been enabled, it may take some time while inventory is being collected on the VM before data appears.

## Track changes

On your VM, select **Change Tracking** under **OPERATIONS**. Select **Edit Settings**, the **Change Tracking** page is displayed. Select the type of setting you want to track and then select **+ Add** to configure the settings. The available option Linux is **Linux Files**

For detailed information on Change Tracking see, [Troubleshoot changes on a VM](#)

## View inventory

On your VM, select **Inventory** under **OPERATIONS**. On the **Software** tab, there is a table list the software that had been found. The high-level details for each software record are viewable in the table. These details include the software name, version, publisher, last refreshed time.

The screenshot shows the 'Software' tab of the 'Inventory' configuration page. At the top, there is a summary: 'New software 1' (with a '7' icon), 'Last 24 hours', and links to 'Learn more', 'Inventory', and 'Provide feedback'. Below this is a navigation bar with tabs: Software (which is selected and highlighted in blue), Files, and Linux Daemons. There is also a search bar: 'Search to filter items...'. The main area is a table with the following columns: NAME, VERSION, PUBLISHER, and LAST REFRESHED TIME. The table lists the following software packages:

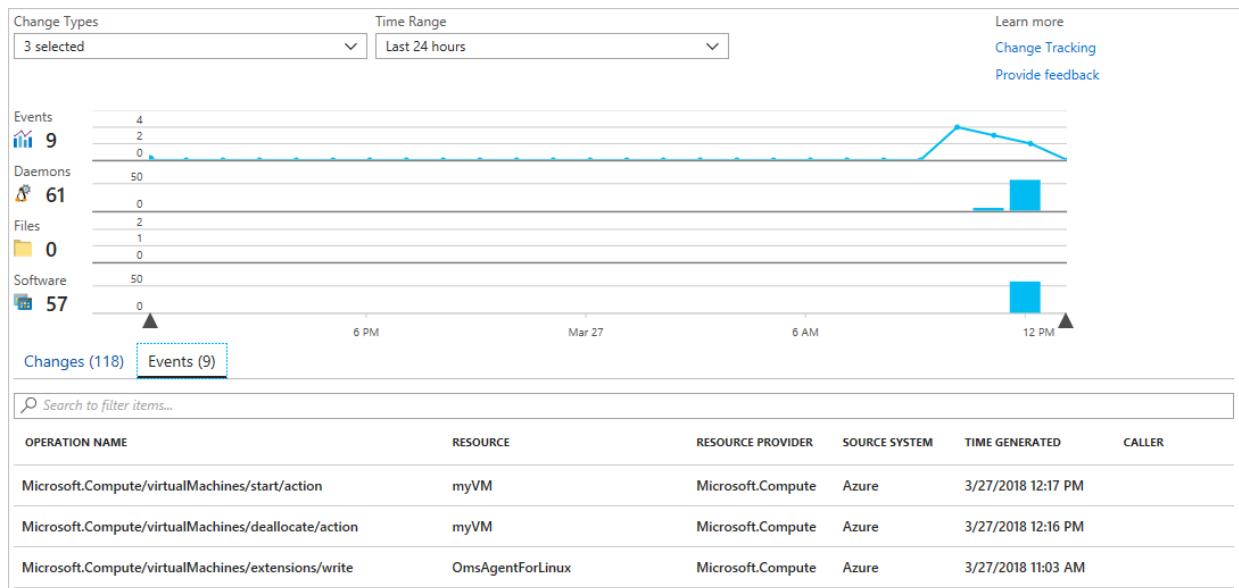
NAME	VERSION	PUBLISHER	LAST REFRESHED TIME	...
accountsservice	0.6.40-2ubuntu11.3	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...
acl	2.2.52-3	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...
acpid	1:2.0.26-1ubuntu2	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...
adduser	3.113+nmu3ubuntu4	Ubuntu Core Developers <...	3/27/2018 11:51 AM	...
apparmor	2.10.95-0ubuntu2.8	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...
apport	2.20.1-0ubuntu2.15	Martin Pitt <martin.pitt@u...	3/27/2018 11:51 AM	...
apport-symptoms	0.20	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...
apt	1.2.25	Ubuntu Developers <ubun...	3/27/2018 11:51 AM	...

## Monitor Activity logs and changes

From the **Change tracking** page on your VM, select **Manage Activity Log Connection**. This task opens the **Azure Activity log** page. Select **Connect** to connect Change tracking to the Azure activity log for your VM.

With this setting enabled, navigate to the **Overview** page for your VM and select **Stop** to stop your VM. When prompted, select **Yes** to stop the VM. When it is deallocated, select **Start** to restart your VM.

Stopping and starting a VM logs an event in its activity log. Navigate back to the **Change tracking** page. Select the **Events** tab at the bottom of the page. After a while, the events shown in the chart and the table. Each event can be selected to view detailed information on the event.



The chart shows changes that have occurred over time. After you have added an Activity Log connection, the line graph at the top displays Azure Activity Log events. Each row of bar graphs represents a different trackable Change type. These types are Linux daemons, files, and software. The change tab shows the details for the changes shown in the visualization in descending order of time that the change occurred (most recent first).

## Next steps

In this tutorial, you configured and reviewed Change Tracking and Update Management for your VM. You learned how to:

- Create a resource group and VM
- Manage Linux updates
- Monitor changes and inventory

Advance to the next tutorial to learn about monitoring your VM.

[Monitor virtual machines](#)

# Tutorial: Monitor changes and update a Windows virtual machine in Azure

9/21/2022 • 9 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

With Azure [Change Tracking](#) and [Update Management](#), you can easily identify changes in your Windows virtual machines in Azure and manage operating system updates for those VMs.

In this tutorial, you learn how to:

- Manage Windows updates.
- Monitor changes and inventory.

## Open Azure Cloud Shell

Azure Cloud Shell is a free interactive shell that you can use to run the steps in this article. It has common Azure tools preinstalled and configured to use with your Azure account.

To open any code block in Cloud Shell, just select Try it from the upper-right corner of that code block.

You can also open Cloud Shell in a separate browser tab by going to <https://shell.azure.com/powershell>. Select Copy to copy code blocks, paste them into the Cloud Shell tab, and select the Enter key to run the code.

## Create a virtual machine

To configure Azure monitoring and update management in this tutorial, you need a Windows VM in Azure.

First, set an administrator username and password for the VM with [Get-Credential](#):

```
$cred = Get-Credential
```

Next, create the VM with [New-AzVM](#). The following example creates a VM named myVM in the East US location. If they don't already exist, the resource group myResourceGroupMonitor and supporting network resources are created:

```
New-AzVm ` 
    -ResourceGroupName "myResourceGroupMonitor" ` 
    -Name "myVM" ` 
    -Location "East US" ` 
    -Credential $cred
```

It takes a few minutes for the resources and VM to be created.

## Manage Windows updates

Update Management helps you manage updates and patches for your Azure Windows VMs. Directly from your VM, you can quickly:

- Assess the status of available updates.
- Schedule installation of required updates.

- Review deployment results to verify updates were successfully applied to the VM.

For pricing information, see [Automation pricing for Update management](#).

## Enable Update Management

To enable Update Management for your VM:

1. Navigate to your VM in the Azure portal (search for **Virtual machines** in the search bar, then choose a VM from the list).
2. Select **Updates** under Operations.
3. Click on **Go to Updates using automation**.
4. The **Enable Update Management** window opens.

Validation is done to determine if Update Management is enabled for this VM. Validation includes checks for a Log Analytics workspace, for a linked Automation account, and for whether the solution is in the workspace.

You use a [Log Analytics](#) workspace to collect data that is generated by features and services such as Update Management. The workspace provides a single location to review and analyze data from multiple sources.

To perform additional actions on VMs that require updates, you can use Azure Automation to run runbooks against VMs. Such actions include downloading or applying updates.

The validation process also checks to see if the VM is provisioned with the Microsoft Monitoring Agent (MMA) and Automation Hybrid Runbook Worker. You use the agent to communicate with the VM and obtain information about the update status.

In the **Enable Update Management** window, choose the Log Analytics workspace and automation account, and then select **Enable**. The solution takes up to 15 minutes to become enabled.

Any of the following prerequisites that are missing during onboarding are automatically added:

- [Log Analytics](#) workspace
- [Automation](#)
- A [Hybrid runbook worker](#), which is enabled on the VM

After the solution is enabled, the **Update management** window opens. Configure the location, Log Analytics workspace and Automation account to use, and then select **Enable**. If these options appear dimmed, another automation solution is enabled for the VM, and that solution's workspace and Automation account must be used.

The Update Management solution can take up to 15 minutes to become enabled. During this time, don't close the browser window. After the solution is enabled, information about missing updates on the VM flows to Azure Monitor logs. It can take from 30 minutes to 6 hours for the data to become available for analysis.

## View an update assessment

After Update Management is enabled, the **Update management** window appears. After the evaluation of updates is finished, you see a list of missing updates on the **Missing updates** tab.

UPDATE NAME	CLASSIFICATION	PUBLISHED DATE	INFORMATION LINK
2019-05 Cumulative Update for Windows Server 2016 for ...	Updates	5/22/2019	<a href="#">KB4499177</a>
Definition Update for Windows Defender Antivirus - KB22...	Definition updates	6/2/2019	<a href="#">KB2267602</a>
Definition Update for Windows Defender Antivirus - KB22...	Definition updates	6/2/2019	<a href="#">KB2267602</a>

## Schedule an update deployment

To install updates, schedule a deployment that follows your release schedule and service window. You choose which update types to include in the deployment. For example, you can include critical or security updates and exclude update rollups.

To schedule a new update deployment for the VM, select **Schedule update deployment** at the top of the **Update management** window. In the **New update deployment** window, specify the following information:

OPTION	DESCRIPTION
Name	Enter a unique name to identify the update deployment.
Operating system	Select either <b>Linux</b> or <b>Windows</b> .

OPTION	DESCRIPTION
<b>Groups to update</b>	<p>For VMs hosted on Azure, define a query based on a combination of subscription, resource groups, locations, and tags. This query builds a dynamic group of Azure-hosted VMs to include in your deployment.</p> <p>For VMs not hosted on Azure, select an existing saved search. With this search, you can select a group of these VMs to include in the deployment.</p> <p>To learn more, see <a href="#">Dynamic Groups</a>.</p>
<b>Machines to update</b>	<p>Select <b>Saved search</b>, <b>Imported group</b>, or <b>Machines</b>.</p> <p>If you select <b>Machines</b>, you can choose individual machines from the drop-down list. The readiness of each machine is shown in the <b>UPDATE AGENT READINESS</b> column of the table.</p> <p>To learn about the different methods of creating computer groups in Azure Monitor logs, see <a href="#">Computer groups in Azure Monitor logs</a></p>
<b>Update classifications</b>	Choose all necessary update classifications.
<b>Include/exclude updates</b>	Select this option to open the <b>Include/Exclude</b> pane. Updates to be included and those to be excluded are on separate tabs. For more information on how inclusion is handled, see <a href="#">Schedule an Update Deployment</a> .
<b>Schedule settings</b>	Choose the time to start, and select either <b>Once</b> or <b>Recurring</b> .
<b>Pre-scripts + Post-scripts</b>	Choose the scripts to run before and after your deployment.
<b>Maintenance window</b>	Enter the number of minutes set for updates. Valid values range from 30 to 360 minutes.
<b>Reboot control</b>	<p>Select how reboots are handled. Available selections are:</p> <ul style="list-style-type: none"> <li>• <b>Reboot if required</b></li> <li>• <b>Always reboot</b></li> <li>• <b>Never reboot</b></li> <li>• <b>Only reboot</b></li> </ul> <p><b>Reboot if required</b> is the default selection. If you select <b>Only reboot</b>, updates aren't installed.</p>

After you have finished configuring the schedule, click **Create** to return to the status dashboard. The **Scheduled** table shows the deployment schedule you created.

You can also create update deployments programmatically. To learn how to create an update deployment with the REST API, see [Software Update Configurations - Create](#). There's also a sample runbook that you can use to create a weekly update deployment. To learn more about this runbook, see [Create a weekly update deployment for one or more VMs in a resource group](#).

### View results of an update deployment

After the scheduled deployment starts, you can see the deployment status in the **Update deployments** tab of the **Update management** window.

If the deployment is currently running, its status shows as "In progress." After successful completion, the status changes to "Succeeded." But if any updates in the deployment fail, the status is "Partially failed."

Select the completed update deployment to see the dashboard for that deployment.

UPDATE NAME	STATUS
2019-05 Cumulative Update for Windows Server 2016 for x64-based Systems (KB4505052)	Succeeded
Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.293.1980.0)	Succeeded
2019-05 Servicing Stack Update for Windows Server 2016 for x64-based Systems (KB4498947)	Succeeded
Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.293.1982.0)	Succeeded
Windows Malicious Software Removal Tool x64 - May 2019 (KB890830)	Succeeded

The **Update results** tile shows a summary of the total number of updates and deployment results on the VM. The table to the right shows a detailed breakdown of each update and the installation results. Each result has one of the following values:

- **Not attempted:** The update isn't installed. There wasn't enough time available based on the defined maintenance-window duration.
- **Succeeded:** The update succeeded.
- **Failed:** The update failed.

Select **All logs** to see all log entries that the deployment created.

Select the **Output** tile to see the job stream of the runbook responsible for managing the update deployment on the target VM.

Select **Errors** to see detailed information about any deployment errors.

## Monitor changes and inventory

You can collect and view an inventory of the software, files, Linux daemons, Windows services, and Windows registry keys on your computers. Tracking the configurations of your machines helps you pinpoint operational issues across your environment and better understand the state of your machines.

### Enable change and inventory management

To enable change and inventory management for your VM:

1. On the leftmost side of the window, select **Virtual machines**.
2. Choose a VM from the list.
3. Under **Operations** in the VM window, select either **Inventory** or **Change tracking**.
4. The **Enable Change Tracking and Inventory** pane opens.

Configure the location, Log Analytics workspace, and Automation account to use, and then select **Enable**. If the options appear dimmed, an automation solution is already enabled for the VM. In that case, the already enabled workspace and Automation account must be used.

Even though the solutions appear separately in the menu, they're the same solution. Enabling one enables both for your VM.

The screenshot shows the Azure portal interface for managing a virtual machine named 'myVM'. On the left, there's a sidebar with various operational services: Auto-shutdown, Backup, Disaster recovery (Preview), Update management, Inventory (which is selected and highlighted in blue), and Change tracking. Under MONITORING, there are options for Metrics, Alert rules, Diagnostics settings, and Advisor recommendations. The main pane is titled 'Inventory' and contains descriptive text about enabling consistent control and compliance using Change Tracking and Inventory. It mentions that this service is included with Azure virtual machines and requires a Log Analytics workspace and an Automation account. Below this, there are dropdown menus for 'Location' (set to 'East US'), 'Log Analytics workspace' (set to 'defaultworkspace'), and 'Automation account' (set to 'Automate'). A large blue 'Enable' button is located at the bottom of the configuration area.

After the solution has been enabled, it might take some time for inventory to be collected on the VM before data appears.

## Track changes

On your VM under **OPERATIONS**, select **Change Tracking** and then select **Edit Settings**. The **Change Tracking** pane opens. Select the type of setting you want to track and then select **+ Add** to configure the settings.

The available settings options for Windows are:

- Windows Registry
- Windows Files

For detailed information on Change Tracking, see [Troubleshoot changes on a VM](#).

## View inventory

On your VM select **Inventory** under **OPERATIONS**. On the **Software** tab, there's a table that shows the software that had been found. The high-level details for each software record appear in the table. These details include the software name, version, publisher, and last refreshed time.

The screenshot shows the 'Software' tab for the 'Inventory' service in the Azure portal. At the top, there are links for 'New software', 'Learn more', 'Inventory', and 'Provide feedback'. Below this, a message says 'Last 24 hours' with a '0' icon. There are tabs for 'Software', 'Files', 'Windows Registry', and 'Windows Services', with 'Software' being the active tab. A search bar is present. The main area is a table with columns: NAME, VERSION, PUBLISHER, and LAST REFRESHED TIME. The table lists several software packages, all of which are 'Installed' and published by Microsoft Corporation, with the last refresh time being 3/22/2018, 9:08 AM.

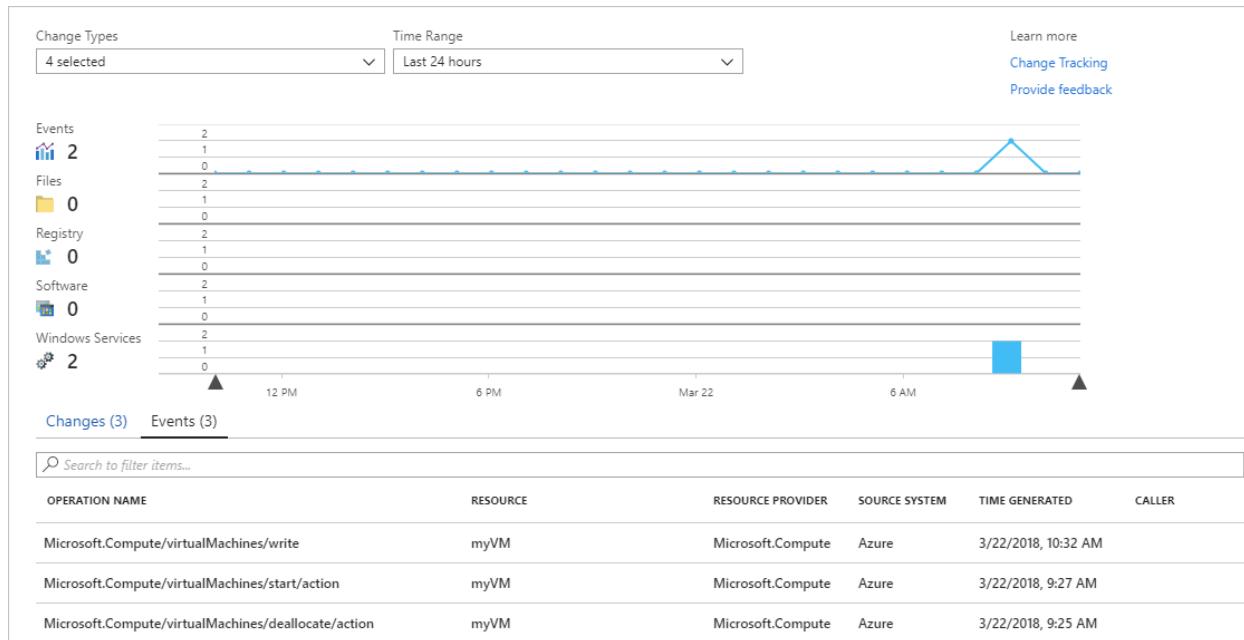
NAME	VERSION	PUBLISHER	LAST REFRESHED TIME
2018-03 Cumulative Update for Windows Server 2016 for x64-based Systems (KB4088787)	Installed	Microsoft Corporation	3/22/2018, 9:08 AM
Definition Update for Windows Defender Antivirus - KB2267602 (Definition 1.263.954.0)	Installed	Microsoft Corporation	3/22/2018, 9:08 AM
Microsoft Monitoring Agent	8.0.11081.0	Microsoft Corporation	3/22/2018, 9:08 AM
Update for Windows Defender antimalware platform - KB4052623 (Version 4.12.17007.18022)	Installed	Microsoft Corporation	3/22/2018, 9:08 AM
Windows Malicious Software Removal Tool x64 - March 2018 (KB890830)	Installed	Microsoft Corporation	3/22/2018, 9:08 AM

## Monitor activity logs and changes

From the **Change tracking** window on your VM, select **Manage Activity Log Connection** to open the **Azure Activity log** pane. Select **Connect** to connect Change Tracking to the Azure activity log for your VM.

After Change Tracking is enabled, go to the **Overview** pane for your VM and select **Stop** to stop your VM. When prompted, select **Yes** to stop the VM. After the VM is deallocated, select **Start** to restart your VM.

Stopping and restarting a VM logs an event in its activity log. Go back to the **Change tracking** pane and select the **Events** tab at the bottom of the pane. After a while, the events appear in the chart and the table. You can select each event to view detailed information for that event.



The previous chart shows changes that have occurred over time. After you add an Azure Activity Log connection, the line graph at the top displays Azure Activity Log events.

Each row of bar graphs represents a different trackable change type. These types are Linux daemons, files, Windows registry keys, software, and Windows services. The **Change** tab shows the change details. Changes appear in the order of when each occurred, with the most recent change shown first.

## Next steps

In this tutorial, you configured and reviewed Change Tracking and Update Management for your VM. You learned how to:

- Create a resource group and VM.
- Manage Windows updates.
- Monitor changes and inventory.

Go to the next tutorial to learn about monitoring your VM.

[Monitor virtual machines](#)

# Monitor Azure virtual machines

9/21/2022 • 6 minutes to read • [Edit Online](#)

When you have critical applications and business processes that rely on Azure resources, it's important to monitor those resources for their availability, performance, and operation. This article describes the monitoring data that's generated by Azure virtual machines (VMs), and it discusses how to use the features of [Azure Monitor](#) to analyze and alert you about this data.

## NOTE

This article provides basic information to help you get started with monitoring your VMs. For a complete guide to monitoring your entire environment of Azure and hybrid virtual machines, see [Monitor virtual machines with Azure Monitor](#).

## What is Azure Monitor?

[Azure Monitor](#) is a full stack monitoring service that provides a complete set of features to monitor your Azure resources. You don't need to directly interact with Azure Monitor, though, to perform a variety of monitoring tasks, because its features are integrated with the Azure portal for the Azure services that it monitors. For a tutorial with an overview of how Azure Monitor works with Azure resources, see [Monitor Azure resources by using Azure Monitor](#).

## Monitoring virtual machine data

Azure virtual machines collect the same kinds of monitoring data as other Azure resources, which are described in [Monitoring data from Azure resources](#). For detailed information about the metrics and logs that are created by Azure virtual machines, see [Reference: Monitoring Azure virtual machine data](#).

## Overview page

To begin exploring Azure Monitor, go to the [Overview](#) page for your virtual machine, and then select the [Monitoring](#) tab. You can see the number of active alerts on the tab.

The [Alerts](#) pane shows you the alerts fired in the last 24 hours, along with important statistics about those alerts. If there are no alerts configured for your VM, there is a link to help you quickly create new alerts for your VM.

**CH1-RETAILVM01** ⚡ ...

Virtual machine | Directory: contosohotels.com

Search (Ctrl+ /) Connect Start Restart Stop Capture Delete Refresh Open in mobile CLI / PS Feedback

Diagnose and solve problems

Monitoring

- Diagnostic settings
- Support + troubleshooting
- Boot diagnostics
- Performance diagnostics

Resource group (move) : CH1-FABRIKAMRG

Status : Running

Location : East US

Subscription (move) : Contoso Hotels Tenant - Production

Subscription ID : ebb79bc0-aa86-44a7-8111-cabbe0c43993

Tags (edit) : updateDomain : 0

Operating system : Windows (Windows Server 2016 Datacenter)

Size : Standard D2s v3 (2 vcpus, 8 GiB memory)

Public IP address : 52.151.200.50

Virtual network/subnet : CH1-RetailAppVNET/default

DNS name : ch1-retailvm01v37ha6.eastus.cloudapp.azure.com

View Cost | JSON View

Properties Monitoring (65 alerts) Capabilities (8) Recommendations (14) Tutorials

**Alerts (65)**

Total fired alerts (last 24h)	Critical	Error	Warning	Informational	Verbose
! 65	0	60	0	0	5

Name ↑ Severity ↑ User response ↑ Fired time ↑

High CPU on production VM [Metric al...   1 - Error	New	3/17/2022, 9:47 AM
High CPU alert monitoring all VMs in s...   1 - Error	New	3/17/2022, 9:46 AM
Anomalous CPU usage on production ...   1 - Error	New	3/17/2022, 9:44 AM
High CPU on production VM [Metric al...   1 - Error	New	3/17/2022, 9:41 AM
High CPU alert monitoring all VMs in s...   1 - Error	New	3/17/2022, 9:40 AM

**Key Metrics** See all metrics

Show data for last: 1 hour 6 hours 12 hours 1 day 7 days 30 days

**CPU (average)**

Percentage CPU Avg  
ch1-retailvm01  
6.4525 %

**Network (total)**

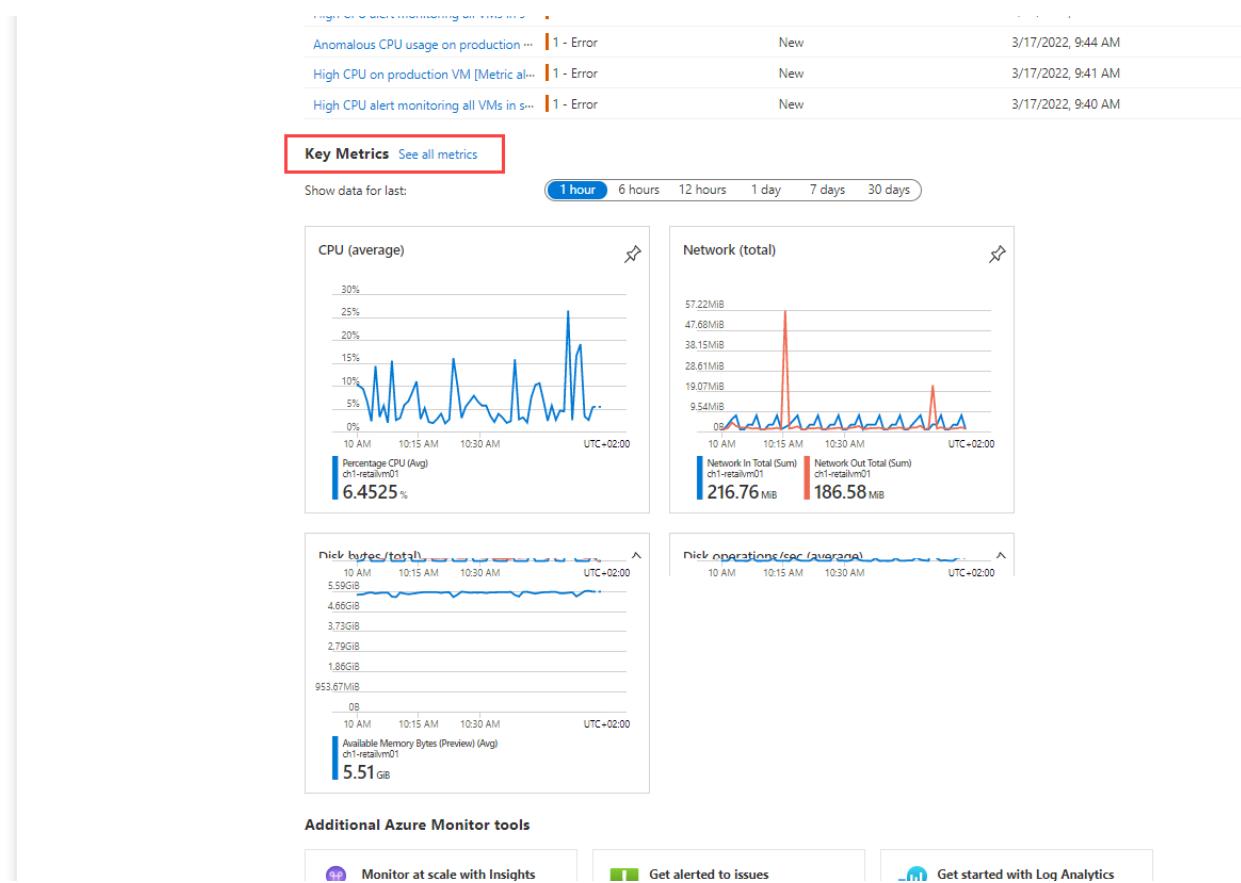
Network In Total (Sum)  
ch1-retailvm01  
216.76 MB

Network Out Total (Sum)  
ch1-retailvm01  
186.58 MB

**nick bytes/total**

**nick operations/sec (average)**

The **Key Metrics** pane includes charts that show key health metrics, such as average CPU and network utilization. At the top of the pane, you can select a duration to change the time range for the charts, or select a chart to open the **Metrics** pane to drill down further or to create an alert rule.



## Activity log

The [Activity log](#) displays recent activity by the virtual machine, including any configuration changes and when it was stopped and started. View the Activity log in the Azure portal, or create a [diagnostic setting to send it to a Log Analytics workspace](#), where you can view events over time or analyze them with other collected data.

The screenshot shows the Azure portal's Activity log for a virtual machine named "srv-win-01". The left sidebar includes options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Networking, Connect, Windows Admin Center (preview), Disks, Size, Security, Advisor recommendations, Extensions + applications, Continuous delivery, Availability + scaling, Configuration, Identity, Properties, and Locks.

The main area displays the Activity log with the following details:

- Filtering:** Management Group: None, Subscription: my-subscription, Event severity: All, Timespan: Last 24 hours, Resource: srv-win-01.
- Table Headers:** Operation name, Status, Time, Time stamp, Subscription, Event initiated by.
- Events:** A list of 13 items, including:
  - Start Virtual Machine (Accepted, 2 minutes ago)
  - Start Virtual Machine (Started, 2 minutes ago)
  - Create or Update Virtual Machine (Succeeded, 2 minutes ago)
  - Create or Update Virtual Machine (Started, 2 minutes ago)
  - Append (Succeeded, 2 minutes ago)
  - Create or Update Virtual Machine (Accepted, 2 minutes ago)
  - Deallocate Virtual Machine (Succeeded, 3 minutes ago)
  - Deallocate Virtual Machine (Started, 4 minutes ago)
  - Deallocate Virtual Machine (Accepted, 4 minutes ago)
  - Health Event Resolved (Resolved, 13 hours ago)
  - Start Virtual Machine (Succeeded, 14 hours ago)
  - Health Event Updated (Updated, 14 hours ago)
  - Health Event Updated (Updated, 14 hours ago)

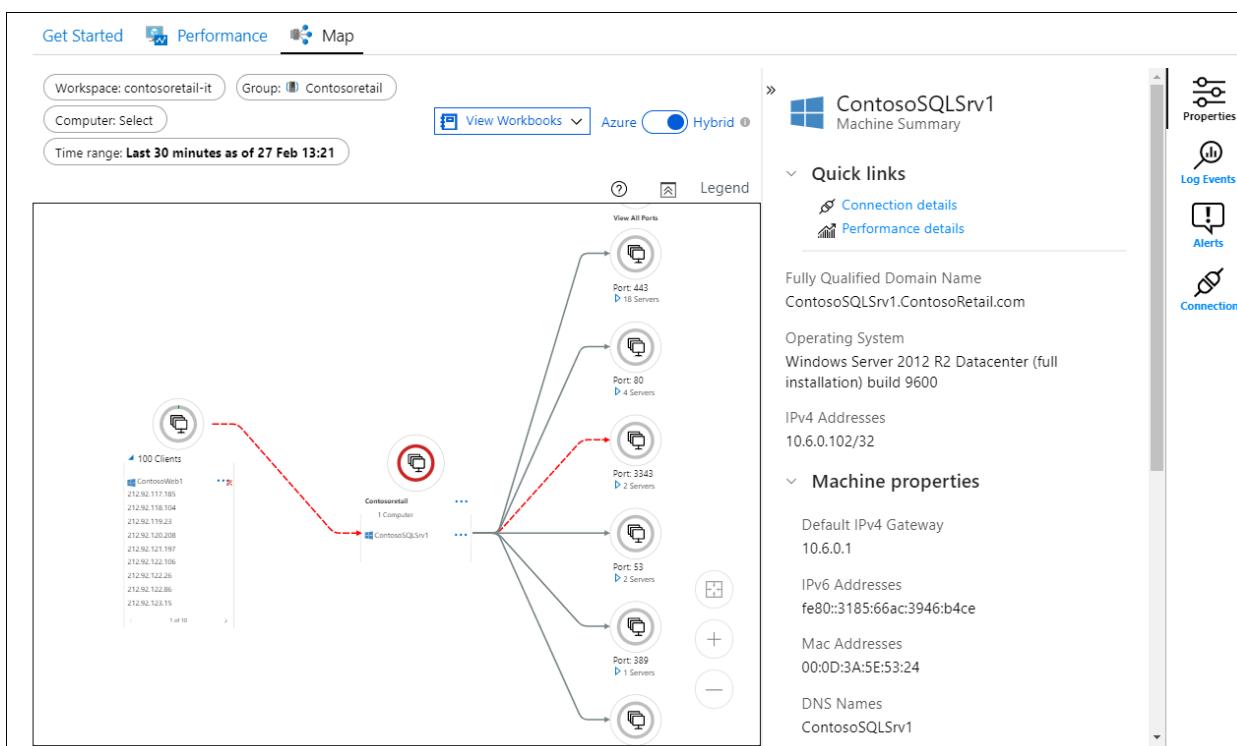
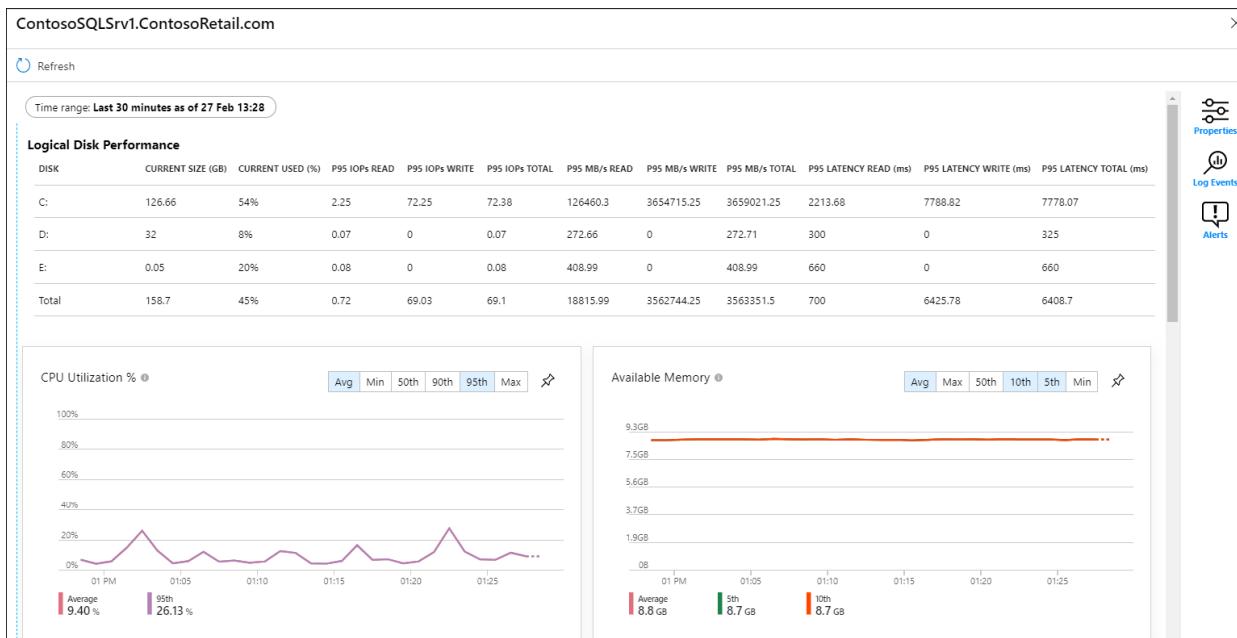
## VM insights

Some services in Azure display customized monitoring experiences in Azure Monitor. These experiences are called *insights*, and they include pre-built workbooks and other specialized features for that particular service.

VM insights are designed to monitor your entire set of Azure and hybrid virtual machines in a single interface.

When you start monitoring multiple virtual machines in your Azure environment, you might want to consider enabling VM insights for the following features:

- Simplified onboarding of the Log Analytics agent and the Dependency agent, so that you can monitor a virtual machine guest operating system and workloads.
- Pre-defined trending performance charts and workbooks, so that you can analyze core performance metrics from the virtual machine's guest operating system.
- The Dependency map, which displays processes that run on each virtual machine and the interconnected components with other machines and external sources.



For quick steps for configuring VM insights and enabling monitoring for a virtual machine, see [Enable Azure Monitor for a single virtual machine or virtual machine scale set in the Azure portal](#).

For general information about enabling insights and a variety of methods for onboarding virtual machines, see [Enable VM insights overview](#).

# Collect guest metrics and logs

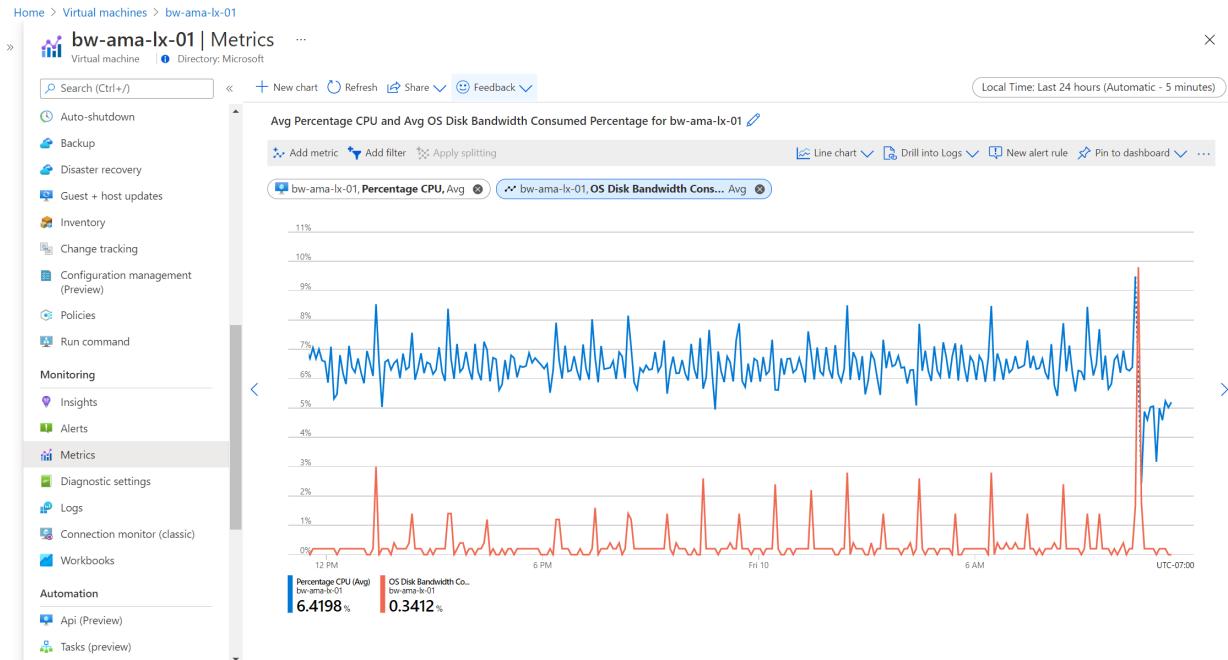
Azure Monitor starts automatically collecting metric data for your virtual machine host when you create the VM. To collect metrics from the guest operating system of the virtual machine, though, you must install an agent. When you enable [VM insights](#), the Log Analytics agent is installed and starts sending performance data to Azure Monitor Logs, which then enables the **Performance** and **Map** views. For a tutorial on enabling VM insights for a virtual machine, see [Enable monitoring for Azure virtual machines](#).

After you've enabled VM insights, install the [Azure Monitor agent](#) so that you can collect guest logs from your virtual machine and send guest metrics to the Azure Monitor **Metrics** pane. By doing so, you can also [analyze metrics on the Metrics pane](#). To learn how to install the Azure Monitor agent and create a data collection rule that defines the data to collect, see [Tutorial: Collect guest logs and metrics from an Azure virtual machine](#).

## Analyze metrics

Metrics are numerical values that describe some aspect of a system at a particular point in time. Although platform metrics for the virtual machine host are collected automatically, you must [install the Azure Monitor agent](#) to collect guest metrics.

The **Overview** pane includes the most common host metrics, and you can access others by using the **Metrics** pane. With this tool, you can create charts from metric values and visually correlate trends. You can also create a metric alert rule or pin a chart to an Azure dashboard. For a tutorial on using this tool, see [Analyze metrics for an Azure resource](#).



For a list of the available metrics, see [Reference: Monitoring Azure virtual machine data](#).

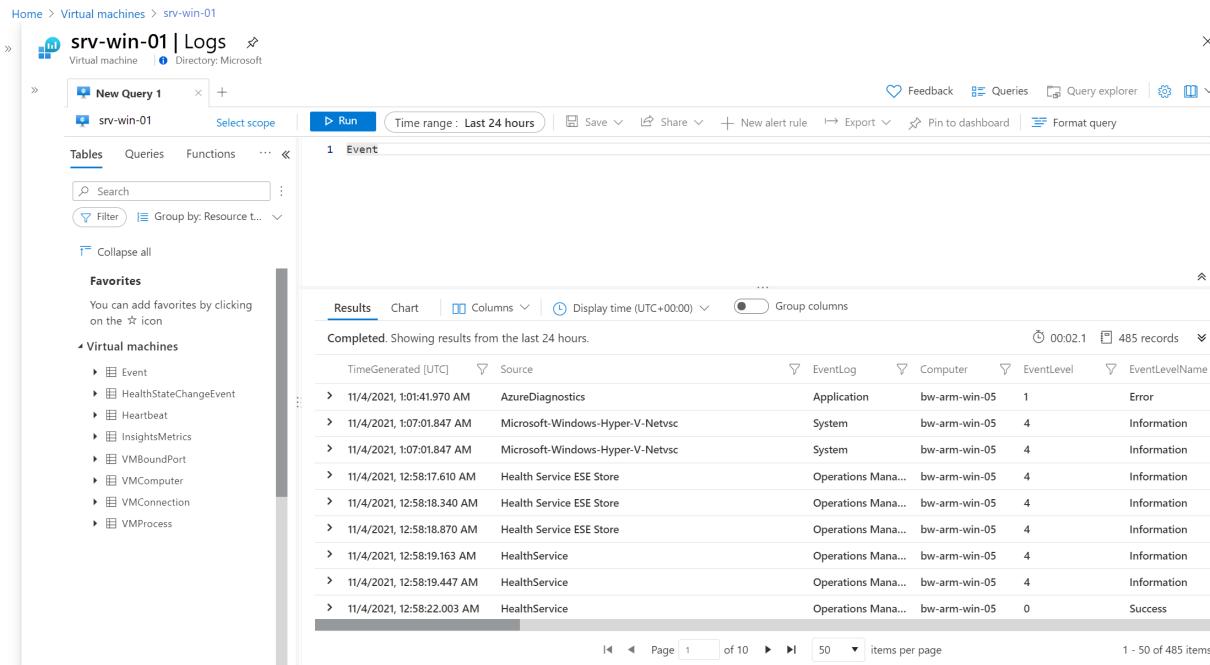
## Analyze logs

Data in Azure Monitor Logs is stored in a Log Analytics workspace, where it's separated into tables, each with its own set of unique properties.

VM insights store the collected data in logs, and the insights provide performance and map views that you can use to interactively analyze the data. You can work directly with this data to drill down further or perform custom analyses. For more information and to get sample queries for this data, see [How to query logs from VM insights](#).

To analyze other log data that you collect from your virtual machines, use [log queries in Log Analytics](#). Several

[built-in queries](#) for virtual machines are available to use, or you can create your own. You can interactively work with the results of these queries, include them in a workbook to make them available to other users, or generate alerts based on their results.



The screenshot shows the Azure Log Analytics interface for a virtual machine named 'srv-win-01'. The left sidebar shows 'Virtual machines' under 'Favorites'. The main area displays a table of events from the last 24 hours. The table has columns: TimeGenerated [UTC], Source, EventLog, Computer, EventLevel, and EventLevelName. The data shows various system events, such as HealthStateChangeEvent, Heartbeat, InsightsMetrics, and VM-related events, with levels ranging from Error to Success.

TimeGenerated [UTC]	Source	EventLog	Computer	EventLevel	EventLevelName
11/4/2021, 1:01:41.970 AM	AzureDiagnostics	Application	bw-arm-win-05	1	Error
11/4/2021, 1:07:01.847 AM	Microsoft-Windows-Hyper-V-Netvsc	System	bw-arm-win-05	4	Information
11/4/2021, 1:07:01.847 AM	Microsoft-Windows-Hyper-V-Netvsc	System	bw-arm-win-05	4	Information
11/4/2021, 12:58:17.610 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:18.340 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:18.870 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:19.163 AM	HealthService	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:19.447 AM	HealthService	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:22.003 AM	HealthService	Operations Mana...	bw-arm-win-05	0	Success

## Alerts

Azure Monitor alerts proactively notify you when important conditions are found in your monitoring data. These alerts can help you identify and address issues in your system before your customers notice them. You can set alerts on [metrics](#), [logs](#), and the [activity log](#).

For more information about the various alerts for Azure virtual machines, see the following resources:

- See [Create an alert when an Azure virtual machine is unavailable](#) for a tutorial on creating a log query alert for when a virtual machine is unresponsive.
- See [Monitor virtual machines with Azure Monitor: Alerts](#) for common alert rules for virtual machines.
- See [Monitor virtual machines with Azure Monitor: Workloads](#) for data you can collect from VM workloads that you can use to create alerts.
- See [Create a log query alert for an Azure resource](#) for a tutorial on creating a log query alert rule.
- For common log alert rules, go to the [Queries](#) pane in Log Analytics. For **Resource type**, enter **Virtual machines**, and for **Type**, enter **Alerts**.

## Next steps

For documentation about the logs and metrics that are generated by Azure virtual machines, see [Reference: Monitoring Azure virtual machine data](#).

# Reference: Monitoring Azure virtual machine data

9/21/2022 • 2 minutes to read • [Edit Online](#)

For more information about collecting and analyzing monitoring data for Azure virtual machines (VMs), see [Monitoring Azure virtual machines](#).

## Metrics

This section lists the platform metrics that are collected for Azure virtual machines and virtual machine scale sets.

METRIC TYPE	RESOURCE PROVIDER / TYPE NAMESPACE AND LINK TO INDIVIDUAL METRICS
Virtual machines	<a href="#">Microsoft.Compute/virtualMachines</a>
Virtual machine scale sets	<a href="#">Microsoft.Compute/virtualMachineScaleSets</a>
Virtual machine scale sets and virtual machines	<a href="#">Microsoft.Compute/virtualMachineScaleSets/virtualMachines</a>

For more information, see a list of [platform metrics that are supported in Azure Monitor](#).

## Metric dimensions

For more information about metric dimensions, see [Multi-dimensional metrics](#).

Azure virtual machines and virtual machine scale sets have the following dimensions that are associated with their metrics.

DIMENSION NAME	DESCRIPTION
LUN	Logical unit number
VMName	Used with virtual machine scale sets

## Azure Monitor Logs tables

This section refers to all the Azure Monitor Logs tables that are relevant to virtual machines and virtual machine scale sets and available for query by Log Analytics.

RESOURCE TYPE	NOTES
<a href="#">Virtual machines</a>	
<a href="#">Virtual machine scale sets</a>	

For reference documentation about Azure Monitor Logs and Log Analytics tables, see the [Azure Monitor Logs](#)

[table reference](#).

## Activity log

The following table lists a few example operations that relate to creating virtual machines in the activity log. For a complete list of possible log entries, see [Microsoft.Compute Resource Provider options](#).

OPERATION	DESCRIPTION
Microsoft.Compute/virtualMachines/start/action	Starts the virtual machine
Microsoft.Compute/virtualMachines/restart/action	Deletes a managed cluster
Microsoft.Compute/virtualMachines/write	Creates a new virtual machine or updates an existing one
Microsoft.Compute/virtualMachines/deallocate/action	Powers off the virtual machine and releases the compute resources
Microsoft.Compute/virtualMachines/extensions/write	Creates a new virtual machine extension or updates an existing one
Microsoft.Compute/virtualMachineScaleSets/write	Starts the instances of the virtual machine scale set

For more information about the schema of activity log entries, see [Activity log schema](#).

## See also

For a description of monitoring Azure virtual machines, see [Monitoring Azure virtual machines](#).

# Tutorial: Enable monitoring for Azure virtual machine

9/21/2022 • 3 minutes to read • [Edit Online](#)

To monitor the health and performance of an Azure virtual machine, you need to install an agent to collect data from its guest operating system. VM insights is a feature of Azure Monitor for monitoring the guest operating system and workloads running on Azure virtual machines. When you enable monitoring for an Azure virtual machine, it installs the necessary agents and starts collecting performance, process, and dependency information from the guest operating system.

## NOTE

If you're completely new to Azure Monitor, you should start with [Tutorial: Monitor Azure resources with Azure Monitor](#). Azure virtual machines generate similar monitoring data as other Azure resources such as platform metrics and Activity log. This tutorial describes how to enable additional monitoring unique to virtual machines.

In this tutorial, you learn how to:

- Create a Log Analytics workspace to collect performance and log data from the virtual machine.
- Enable VM insights for the virtual machine which installs the required agents and begins data collection.
- Inspect graphs analyzing performance data collected from the virtual machine.
- Inspect map showing processes running on the virtual machine and dependencies with other systems.

## NOTE

VM insights installs the Log Analytics agent which collects performance data from the guest operating system of virtual machines. It doesn't collect logs from the guest operating system and doesn't send performance data to Azure Monitor Metrics. For this functionality, see [Tutorial: Collect guest logs and metrics from Azure virtual machine](#).

## Prerequisites

To complete this tutorial you need the following:

- An Azure virtual machine to monitor.

## Create a Log Analytics workspace

Log data in Azure Monitor is stored in a Log Analytics workspace. If you already created a workspace in your subscription, then you can use that one. You can also choose to use the default workspace that's created in each Azure subscription.

If you want to create a new Log Analytics, then you can use the following procedure. If you're going to use an existing one, then move on to the next section.

From **All services** in the Azure portal, select **Log Analytics workspaces**.

The screenshot shows the Microsoft Azure portal's search interface. The search bar at the top contains the text "log analytics workspaces". Below the search bar, the "Services" section is expanded, showing several options: Log Analytics workspaces, Activity log, Azure Synapse Analytics (workspaces preview), Workspaces, Logic Apps, and Data Catalog. To the right of the services, there are sections for "Marketplace" (Log Analytics Workspace) and "Documentation" (links to workspace creation and management). A "See all" link is located at the top right of the services list.

Click **Create** to create a new workspace.

This screenshot shows the "Log Analytics workspaces" list page. At the top left is the breadcrumb navigation "Home > Log Analytics workspaces". Below it is a toolbar with buttons for "+ Create", "Open recycle bin", "Manage view", "Refresh", "Export to CSV", "Open query", "Assign tags", and "Feedback". There are also filters for "Subscription", "Resource group", and "Location". A message at the bottom of the toolbar says "Filter for any field...". The main area displays a list of workspaces, with the first item, "Log Analytics workspace", being the one currently selected.

On the **Basics** tab, select a **Subscription**, **Resource group**, and **Region** for the workspace. These do not need to be the same as the resource being monitored. Provide a **Name** that must be globally unique across all Azure Monitor subscriptions.

This screenshot shows the "Create Log Analytics workspace" wizard on the "Basics" tab. The top navigation bar includes "Home > Log Analytics workspaces > Create Log Analytics workspace". Below the tabs "Basics", "Tags", and "Review + Create", the "Basics" tab is active. A callout box provides information about what a Log Analytics workspace is and links to learn more. The "Project details" section asks for a subscription and resource group. The "Subscription" dropdown is set to "AzureMonitor\_Docs" and the "Resource group" dropdown is set to "my-resource-group". The "Instance details" section asks for a name and region. The "Name" input field is filled with "my-workspace" and the "Region" dropdown is set to "East US". At the bottom are buttons for "Review + Create", "Previous", and "Next : Pricing tier >".

Click **Review + Create** to create the workspace.

## Enable monitoring

Select **Insights** from your virtual machine's menu in the Azure portal. If VM insights hasn't yet been enabled for it, you should see a screen similar to the following allowing you to enable monitoring. Click **Enable**.

### NOTE

If you selected the option to **Enable detailed monitoring** when you created your virtual machine, VM insights may already be enabled. Select your workspace and click **Enable** again. This is the workspace where data collected by VM insights will be sent.

t

## Get more visibility into the health and performance of your virtual machine

With an Azure virtual machine you get host CPU, disk and up/down state of your VMs out of the box. Enabling additional monitoring capabilities provides insights into the performance and dependencies for your virtual machines.

You will be billed based on the amount of data ingested and your data retention settings. It can take between 5-10 minutes to configure the virtual machine and the monitoring data to appear.



The VM is not connected to any workspace. Please select the monitoring workspace where you will store your data

Workspace Subscription \* ⓘ

AzureMonitor\_Docs

Choose a Log Analytics Workspace ⓘ

DefaultWorkspace-00000000-0000-0000-0000-000000000000-EUS [eastus]

Note: If the virtual machine already has either SCOM or OMS agent installed locally, the Microsoft Monitoring Agent (MMA) extension will still be installed and connected to the configured workspace.

The map data set collected with Azure Monitor for VMs is intended to be infrastructure data about the resources being deployed and monitored. For details on data collected please [click here](#).

**Enable**

⚠️ Having difficulties enabling Azure Monitors for VM? [Troubleshoot](#)

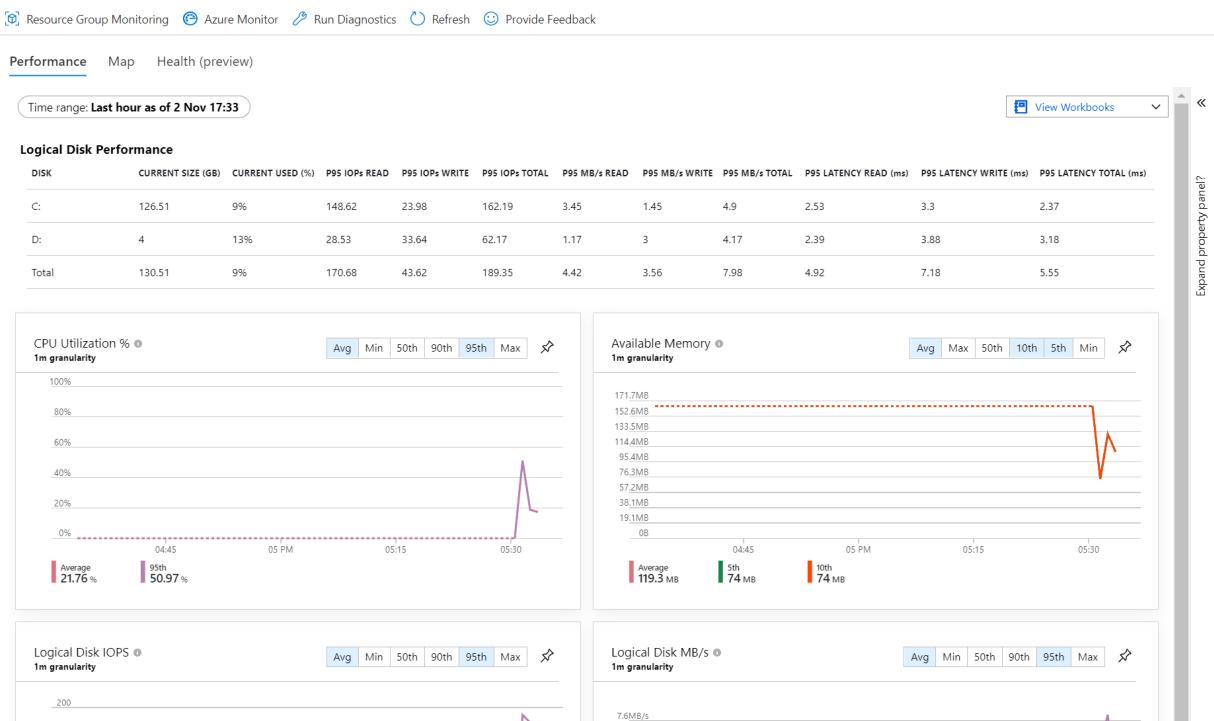
You'll see a message saying that monitoring is being enabled. It may take several minutes for the agent to be installed and for data collection to begin.

### NOTE

You may receive a message about an upgrade being available for VM insights. If so, select the option to perform the upgrade before proceeding.

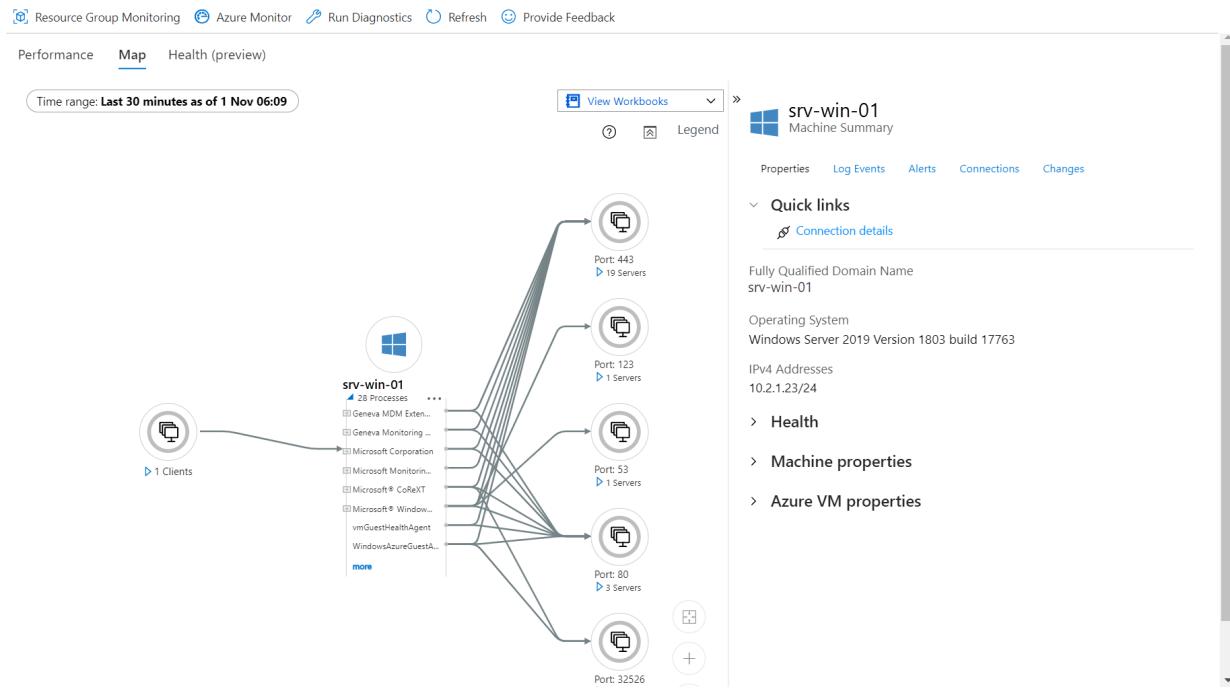
## View performance

When the deployment is complete, you'll see views in the **Performance** tab in VM insights with performance data for the machine. This shows you the values of key guest metrics over time.



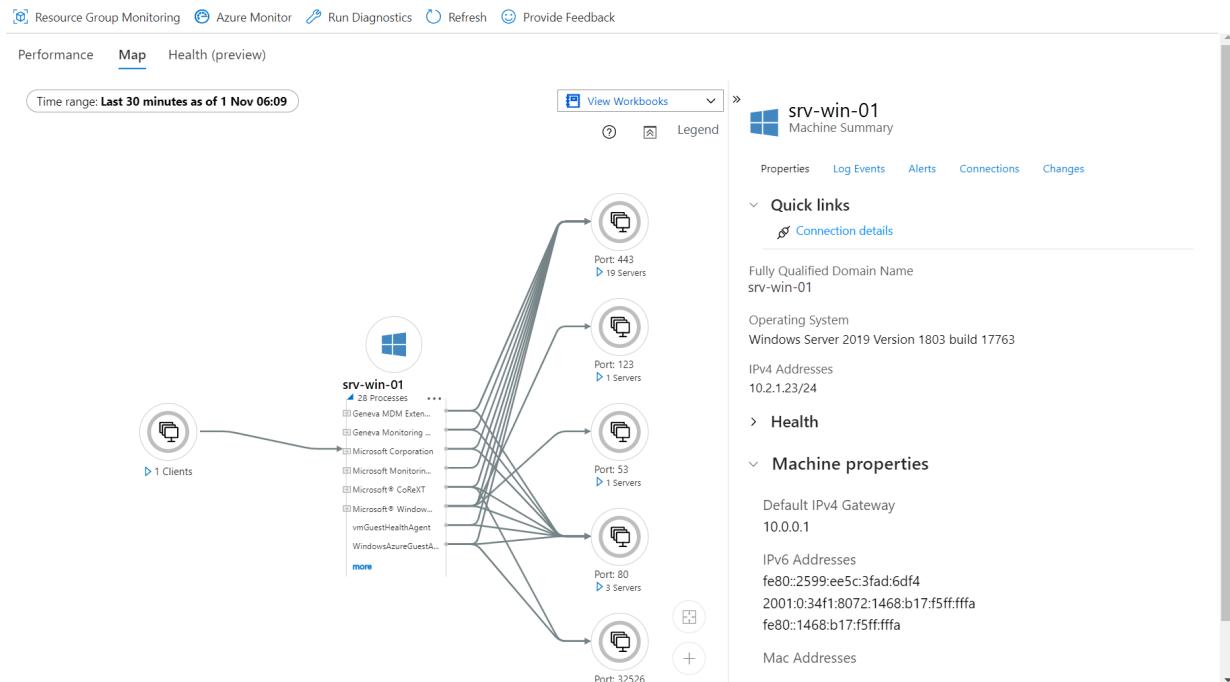
# View processes and dependencies

Select the **Maps** tab to view processes and dependencies for the virtual machine. The current machine is at the center of the view. View the processes running on it by expanding **Processes**.



## View machine details

The **Maps** view provides different tabs with information collected about the virtual machine. Click through the tabs to see what's available.



## Next steps

Now that you're collecting data from the virtual machine, you can use that data to create alerts to proactively notify you when issues are detected.

[Create alert when Azure virtual machine is unavailable](#)

# Tutorial: Create alert when Azure virtual machine is unavailable

9/21/2022 • 4 minutes to read • [Edit Online](#)

One of the most common alerting conditions for a virtual machine is whether the virtual machine is running. Once you enable monitoring with VM insights in Azure Monitor for the virtual machine, a heartbeat is sent to Azure Monitor every minute. You can create a log query alert rule that sends an alert if a heartbeat isn't detected. This method not only alerts if the virtual machine isn't running, but also if it's not responsive.

In this tutorial, you learn how to:

- View log data collected by VM insights in Azure Monitor for a virtual machine.
- Create an alert rule from log data that will proactively notify you if the virtual machine is unavailable.

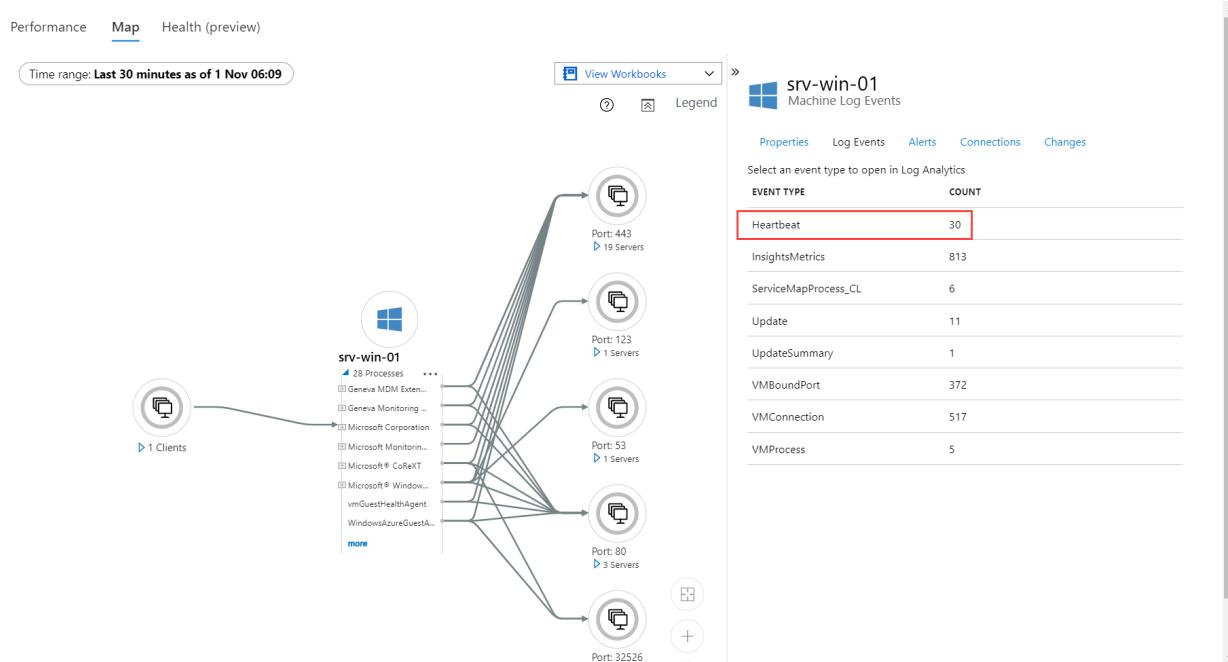
## Prerequisites

To complete this tutorial you need the following:

- An Azure virtual machine to monitor.
- Monitoring with VM insights enabled for the virtual machine. See [Tutorial: Enable monitoring for Azure virtual machine](#).

## Create a heartbeat query

There are multiple ways to create a log query alert rule. For this tutorial, we'll start from the **Logs Events** tab in the **Map** view. This gives a summary of the log data that's been collected for the virtual machine.



Click on **Heartbeat**. This opens Log Analytics, which is the primary tool to analyze log data collected from the virtual machine, with a simple query for heartbeat events. If you click on **TimeGenerated** to sort by that column, you can see that a heartbeat is created each minute.

For the alert, you want to return only heartbeat records in the last 5 minutes. If no records are returned, then you can assume the virtual machine is down.

Add a line to the query to filter the results to only records created in the last 5 minutes. This uses the [ago function](#) that subtracts a particular time span from the current time.

```
Heartbeat  
| where Computer == 'computer-name'  
| where TimeGenerated > ago(5m)
```

Click **Run** to see the results of this query, which should now include just the heartbeats in the last 5 minutes.

The screenshot shows the Azure Log Analytics workspace interface. The top navigation bar includes 'Home > Virtual machines > srv-win-01' and a 'Logs' icon. Below the navigation is a search bar with 'New Query \*' and a red box highlighting the 'Run' button. The query editor contains the following code:

```
1 Heartbeat
2 | where Computer == 'srv-win-01'
3 | where TimeGenerated > ago(5m)
```

The results table has columns: TimeGenerated (UTC), SourceComputerId, ComputerIP, Computer, Category, OSType, OSMajorVersion, OSMinorVersion, and Version. The table displays six rows of completed heartbeat data for the 'srv-win-01' computer. The last row is highlighted with a red box.

TimeGenerated (UTC)	SourceComputerId	ComputerIP	Computer	Category	OSType	OSMajorVersion	OSMinorVersion	Version
11/2/2021, 9:12:40.053 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:13:40.060 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:14:40.067 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:15:40.083 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:16:40.090 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18

## Create alert rule

Now that you have the log query, you can create an alert rule that will send an alert when that query doesn't return any records. If no heartbeat records are returned from the last 5 minutes, then we can assume that machine hasn't been responsive in that time.

Click **New alert rule** to create a rule from the current query.

The screenshot shows the Azure Log Analytics workspace for a workspace named "my-workspace". A query titled "New Query 1" is running, displaying the following log entries:

TimeGenerated [UTC]	SourceComputerId	ComputerIP	Computer	Category	OSType	OSMajorVersion	OSMinorVersion	Version
11/2/2021, 9:12:40.053 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:13:40.060 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:14:40.067 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:15:40.083 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18
11/2/2021, 9:16:40.090 PM	19de8c1d-e6b4-4c0f-9904-c198f653...	20.106.171.133	srv-win-01	Direct Agent	Windows	10	0	10.20.18

At the top right of the query editor, there is a red box around the "New alert rule" button.

The alert rule will already have the **Log query** filled in. The **Measurement** is also already correct since we want to count the number of table rows returned from the query. If the number of rows is zero, then we want to create an alert.

The screenshot shows the "Create alert rule (preview)" configuration page. The "Condition" tab is selected. The "Log query" section contains the same query as the previous screenshot:

```
Heartbeat
| where Computer == 'win-srv-01'
| where TimeGenerated > ago(5m)
```

The "Measurement" section shows the following settings:

Measure:	Table rows
Aggregation type:	Count
Aggregation granularity:	5 minutes

At the bottom, there are "Review + create", "Previous", and "Next: Actions >" buttons.

Scroll down to **Alert logic** and change **Operator** to **Equal to** and provide a **Threshold value** of 0. This means that we want to create an alert when no records are returned, or when the record count from the query equals zero.

Home > srv-win-01 > Logs >

### Create alert rule (preview) ...

**Alert logic**

Operator \* ⓘ Equal to

Threshold value \* ⓘ 0

Frequency of evaluation \* ⓘ 5 minutes

Estimated monthly cost \$1.50 (USD)

See final alert query

Advanced options

**Preview**

Select time series ⓘ Time range ⓘ Over the last 6 hours

No data to display

Review + create Previous Next: Actions >

## Configure action group

The **Actions** page allows you to add one or more **action groups** to the alert rule. Action groups define a set of actions to take when an alert is fired such as sending an email or an SMS message.

If you already have an action group, click **Add action group** to add an existing group to the alert rule.

Home > srv-win-01 > Logs >

### Create alert rule (preview) ...

Scope Condition Actions Details \* Tags Review + create

An action group is a set of actions that can be applied to an alert rule. [Learn more](#)

+ Add action groups + Create action group

Action group name	Contains actions
No action group selected yet	

**Add action groups**

Select up to five action groups to attach to this alert rule.

Subscription ⓘ my-subscription

Search

Action group name ↑	Resource group ↑	Contain actions
<input type="checkbox"/> My action group	my-resource-groups	1 Email, 1 SMS message

Review + create Previous Next: Details > Select

If you don't already have an action group in your subscription to select, then click **Create action group** to create a new one. Select a **Subscription** and **Resource group** for the action group and give it an **Action group name** that will appear in the portal and a **Display name** that will appear in email and SMS notifications.

Home > Monitor > Create alert rule >

### Create action group ...

Basics Notifications Actions Tags Review + create

An action group invokes a defined set of notifications and actions when an alert is triggered. [Learn more](#)

**Project details**

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ AzureMonitor\_Docs

Resource group \* ⓘ bw-ama

[Create new](#)

**Instance details**

Action group name \* ⓘ My action group

Display name \* ⓘ My group

This display name is limited to 12 characters

Select **Notifications** and add one or more methods to notify appropriate people when the alert is fired.

Home > Monitor > Create alert rule >

Create action group ...

Basics Notifications Actions Tags Review + create

**Notifications**

Configure the method in which users will be notified when the action group triggers. Select notification types, provide receiver details and add a unique description. This step is optional.

Notification type	Name	Selected
Email/SMS message/Push/Voice	Mail backup	Email
Email/SMS message/Push/Voice	Text administrator	

Please configure the notification by clicking the edit button.

Email

SMS (Carrier charges may apply)  
Country code \* 1

Azure app Push Notifications  
Azure account email

Voice  
Country code 1

Phone number

Enable the common alert schema. [Learn more](#)

Yes  No

OK

Review + create Previous Next: Actions >

## Configure details

The **Details** page allows you to configure different settings for the alert rule.

- **Subscription and Resource group** where the alert rule will be stored. This doesn't need to be in the same resource group as the resource that you're monitoring.
- **Severity** for the alert. The severity allows you to group alerts with a similar relative importance. A severity of **Error** is appropriate for an unresponsive virtual machine.
- Keep the box checked to **Enable alert upon creation**.
- Keep the box checked to **Automatically resolve alerts**. This will automatically resolve the alert when the virtual machine comes back online and heartbeat records are seen again.

Home > srv-win-01 > Logs >

Create alert rule (preview) ...

Scope Condition Actions Details Tags Review + create

**Project details**

Select the subscription and resource group in which to save the alert rule.

Subscription \* my-subscription

Resource group \* my-resource-group

[Create new](#)

**Alert rule details**

Severity \* 1 - Error

Alert rule name \* Virtual machine down

Alert rule description

Region East US

**Advanced options**

Enable upon creation

Automatically resolve alerts (preview)

Mute actions

Check workspace linked storage

Review + create Previous Next: Tags >

Click **Review + create** to create the alert rule.

## View the alert

To test the alert rule, stop the virtual machine. If you configured a notification in your action group, then you

should receive that notification within a few minutes. You'll also see an alert indicated in the summary shown in the **Alerts** page for the virtual machine.

Home > srv-win-01

! srv-win-01 | Alerts

Virtual machine | Directory: Microsoft

Search (Ctrl+ /)

+ Create Alert rules Action groups Action rules (preview) Refresh Feedback

Subscription: my-subscription Resource group: my-rg Time range: Past 24 hours Resource: srv-win-01

Total alerts: 1 Since 11/2/2021, 2:46 PM Smart groups (preview): 7 30.00% Reduction Total alert rules: 0 Enabled 0 Learn more About alerts

Severity	Total alerts	New	Acknowledged	Closed
0 - Critical	0	0	0	0
1 - Error	1	1	0	0
2 - Warning	0	0	0	0
3 - Informational	0	0	0	0
4 - Verbose	0	0	0	0

Click on the **Severity** to see the list of those alerts. Click on the alert itself to view its details.

Home > srv-win-01

! srv-win-01 | Alerts

Virtual machine | Directory: Microsoft

Search (Ctrl+ /)

+ Create Alert rules Action groups Action rules (preview) Refresh Feedback

Subscription: my-subscription Resource group: my-rg Time range: Past 24 hours Resource: srv-win-01

Total alerts: 1 Since 11/2/2021, 2:46 PM Smart groups (preview): 7 30.00% Reduction Total alert rules: 0 Enabled 0 Learn more About alerts

Severity	Total alerts	New	Acknowledged	Closed
0 - Critical	0	0	0	0
1 - Error	1	1	0	0
2 - Warning	0	0	0	0
3 - Informational	0	0	0	0
4 - Verbose	0	0	0	0

## Next steps

Now that you know how to create an alert from log data, collect additional logs and performance data from the virtual machine with a data collection rule.

[Collect guest logs and metrics from Azure virtual machine](#)

# Tutorial: Collect guest logs and metrics from Azure virtual machine

9/21/2022 • 5 minutes to read • [Edit Online](#)

When you [enable monitoring with VM insights](#), it collects performance data using the Log Analytics agent. To collect logs from the guest operating system and to send performance data to Azure Monitor Metrics, install the [Azure Monitor agent](#) and create a [data collection rule](#) (DCR) that defines the data to collect and where to send it.

## NOTE

Prior to the Azure Monitor agent, guest metrics for Azure virtual machines were collected with the [Azure diagnostic extension](#) for Windows (WAD) and Linux (LAD). These agents are still available and can be configured with the **Diagnostic settings** menu item for the virtual machine, but they are in the process of being replaced with Azure Monitor agent.

In this tutorial, you learn how to:

- Create a data collection rule that send guest performance data to Azure Monitor metrics and log events to Azure Monitor Logs.
- View guest metrics in metrics explorer.
- View guest logs in Log Analytics.

## Prerequisites

To complete this tutorial you need the following:

- An Azure virtual machine to monitor.

## Create data collection rule

[Data collection rules](#) in Azure Monitor define data to collect and where it should be sent. When you define the data collection rule using the Azure portal, you specify the virtual machines it should be applied to. The Azure Monitor agent will automatically be installed on any virtual machines that don't already have it.

## NOTE

You must currently install the Azure Monitor agent from **Monitor** menu in the Azure portal. This functionality is not yet available from the virtual machine's menu.

From the **Monitor** menu in the Azure portal, select **Data Collection Rules** and then **Create** to create a new data collection rule.

Home > Monitor

## Monitor | Data Collection Rules

Microsoft

Search (Ctrl+ /)  + Create Manage view Refresh Export to CSV Open query Assign tags Delete Feedback

Containers Networks SQL (preview) Azure Cosmos DB Key Vaults Azure Cache for Redis Azure Data Explorer Clusters Log Analytics workspaces (preview) Azure Stack HCI (preview) Service Bus (preview) ... Insights Hub

Subscription == my-subscription Resource group == all Location == all Add filter

No grouping List view

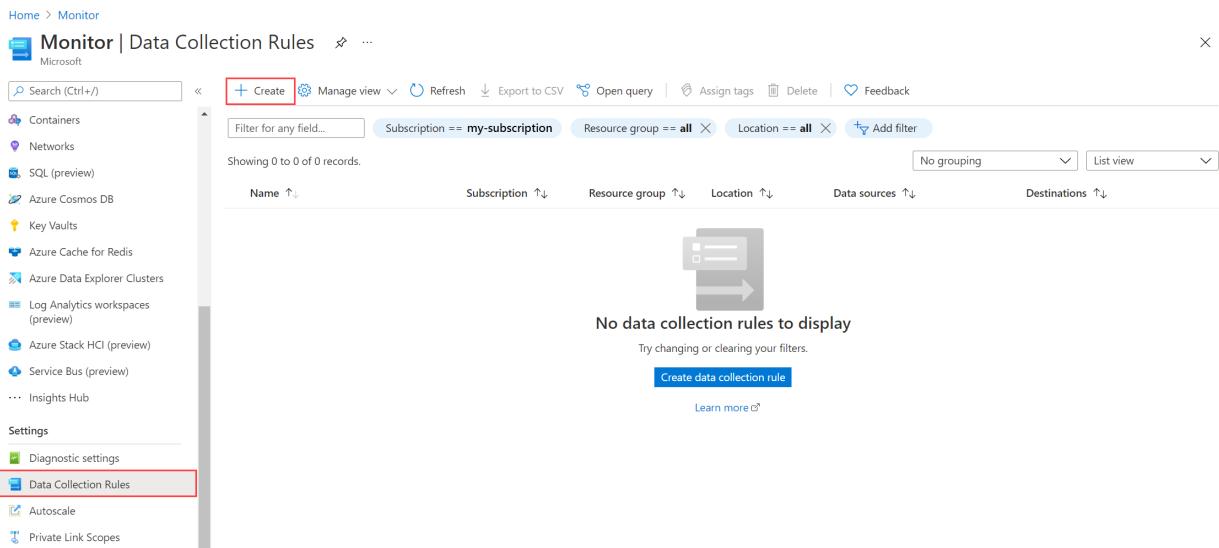
Name ↑ Subscription ↑ Resource group ↑ Location ↑ Data sources ↑ Destinations ↑

No data collection rules to display

Try changing or clearing your filters.

Create data collection rule Learn more

Diagnostic settings Data Collection Rules Autoscale Private Link Scopes



On the **Basics** tab, provide a **Rule Name** which is the name of the rule displayed in the Azure portal. Select a **Subscription**, **Resource Group**, and **Region** where the DCR and its associations will be stored. These do not need to be the same as the resources being monitored. The **Platform Type** defines the options that are available as you define the rest of the DCR. Select *Windows* or *Linux* if it will be associated only those resources or *Custom* if it will be associated with both types.

Home > Monitor >

### Create Data Collection Rule

Data collection rule management

Basics Resources Collect and deliver Review + create

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all of your resources.

Rule details

Rule Name \* collect-logs-metrics

Subscription \* my-subscription

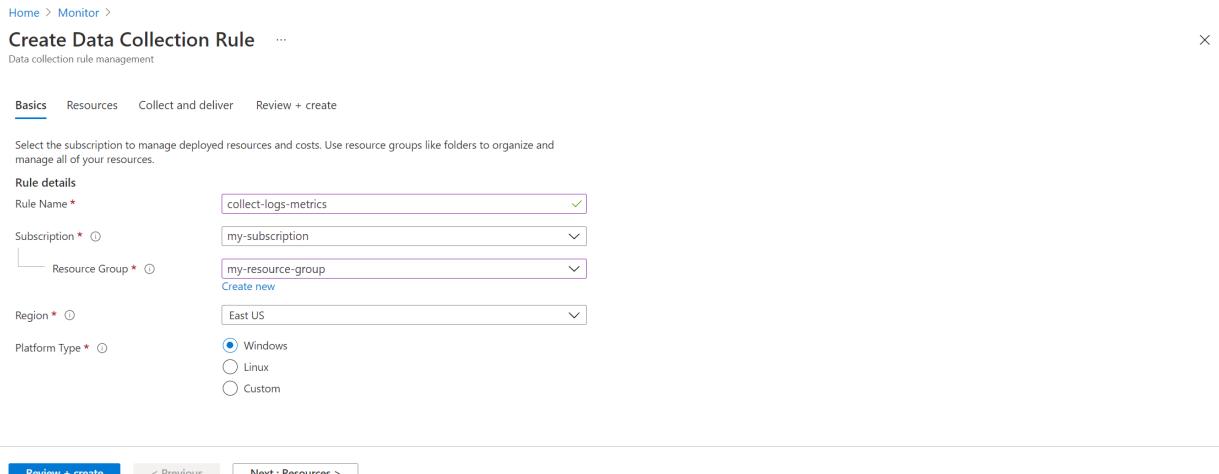
Resource Group \* my-resource-group

Create new

Region \* East US

Platform Type \* Windows

Review + create < Previous Next : Resources >



## Select resources

On the **Resources** tab, identify one or more virtual machines that the data collection rule will apply to. The Azure Monitor agent will be installed on any that don't already have it. Click **Add resources** and select either your virtual machines or the resource group or subscription where your virtual machine is located. The data collection rule will apply to all virtual machines in the selected scope.

## Select data sources

A single data collection rule can have multiple data sources. For this tutorial, we'll use the same rule to collect both guest metrics and guest logs. We'll send metrics to both to Azure Monitor Metrics and to Azure Monitor Logs so that they can be analyzed both with metrics explorer and Log Analytics.

On the **Collect and deliver** tab, click **Add data source**. For the **Data source type**, select **Performance counters**. Leave the **Basic** setting and select the counters that you want to collect. **Custom** allows you to select individual metric values.

Select the **Destination** tab. **Azure Monitor Metrics** should already be listed. Click **Add destination** to add another. Select **Azure Monitor Logs** for the **Destination type**. Select your Log Analytics workspace for the **Account or namespace**. Click **Add data source** to save the data source.

Click **Add data source** again to add logs to the data collection rule. For the **Data source type**, select **Windows event logs** or **Linux syslog**. Select the types of log data that you want to collect.

Select the **Destination** tab. **Azure Monitor Logs** should already be selected for the **Destination type**. Select your Log Analytics workspace for the **Account or namespace**. If you don't already have a workspace, then you can select the default workspace for your subscription, which will automatically be created. Click **Add data source** to save the data source.

Click **Review + create** to create the data collection rule and install the Azure Monitor agent on the selected virtual machines.

Home > Monitor >

## Create Data Collection Rule

Data collection rule management

Basics Resources **Collect and deliver** Review + create

Configure which data sources to collect, and where to send the data to.

+ Add data source

Data source	Destination(s)
Performance counters	Azure Monitor Metrics (preview)
Windows event logs	Azure Monitor Logs
Linux syslog	Azure Monitor Logs

**Review + create** < Previous Next : Review + create >

## Viewing logs

Data is retrieved from a Log Analytics workspace using a log query written in Kusto Query Language (KQL). While a set of pre-created queries are available for virtual machines, we'll use a simple query to have a look at the events that we're collecting.

Select **Logs** from your virtual machine's menu. Log Analytics opens with an empty query window with the scope set to that machine. Any queries will include only records collected from that machine.

### NOTE

The **Queries** window may open when you open Log Analytics. This includes pre-created queries that you can use. For now, close this window since we're going to manually create a simple query.

Home > Virtual machines > srv-win-01

## srv-win-01 | Logs

Virtual machine Directory: Microsoft

New Query 1 + Select scope Run Time range: Last 24 hours Save Share New alert rule Export Pin to dashboard Format query

Tables Queries Functions ...

Search Filter Group by: Resource t...

Collapse all

Favorites You can add favorites by clicking on the icon

Virtual machines

- Event
- HealthStateChangeEvent
- Heartbeat
- InsightsMetrics
- VMBoundPort
- VMComputer
- VMConnection
- VMProcess

Queries History

No queries history

You haven't run any queries yet. To start, go to Queries on the side pane or type a query in the query editor.

In the empty query window, type either **Event** or **Syslog** depending on whether your machine is running Windows or Linux and then click **Run**. The events collected within the **Time range** are displayed.

### NOTE

If the query doesn't return any data, then you may need to wait a few minutes until events are created on the virtual machine to be collected. You may also need to modify the data source in the data collection rule to include additional categories of events.

The screenshot shows the Azure Log Analytics interface for the virtual machine 'srv-win-01'. The left sidebar has a 'Favorites' section with 'Virtual machines' expanded, showing various event types like 'Event', 'HealthStateChangeEvent', 'Heartbeat', etc. The main area is titled 'Logs' and shows a table of events. The table has the following columns: TimeGenerated [UTC], Source, EventLog, Computer, EventLevel, and EventLevelName. The data in the table includes:

TimeGenerated [UTC]	Source	EventLog	Computer	EventLevel	EventLevelName
11/4/2021, 1:01:41.970 AM	AzureDiagnostics	Application	bw-arm-win-05	1	Error
11/4/2021, 1:07:01.847 AM	Microsoft-Windows-Hyper-V-Netvsc	System	bw-arm-win-05	4	Information
11/4/2021, 1:07:01.847 AM	Microsoft-Windows-Hyper-V-Netvsc	System	bw-arm-win-05	4	Information
11/4/2021, 12:58:17.610 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:18.340 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:18.870 AM	Health Service ESE Store	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:19.163 AM	HealthService	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:19.447 AM	HealthService	Operations Mana...	bw-arm-win-05	4	Information
11/4/2021, 12:58:22.003 AM	HealthService	Operations Mana...	bw-arm-win-05	0	Success

For a tutorial on using Log Analytics to analyze log data, see [Log Analytics tutorial](#). For a tutorial on creating alert rules from log data, see [Tutorial: Create a log query alert for an Azure resource](#).

## View guest metrics

You can view metrics for your host virtual machine with metrics explorer without a data collection rule just like [any other Azure resource](#). With the data collection rule though, you can use metrics explorer to view guest metrics in addition to host metrics.

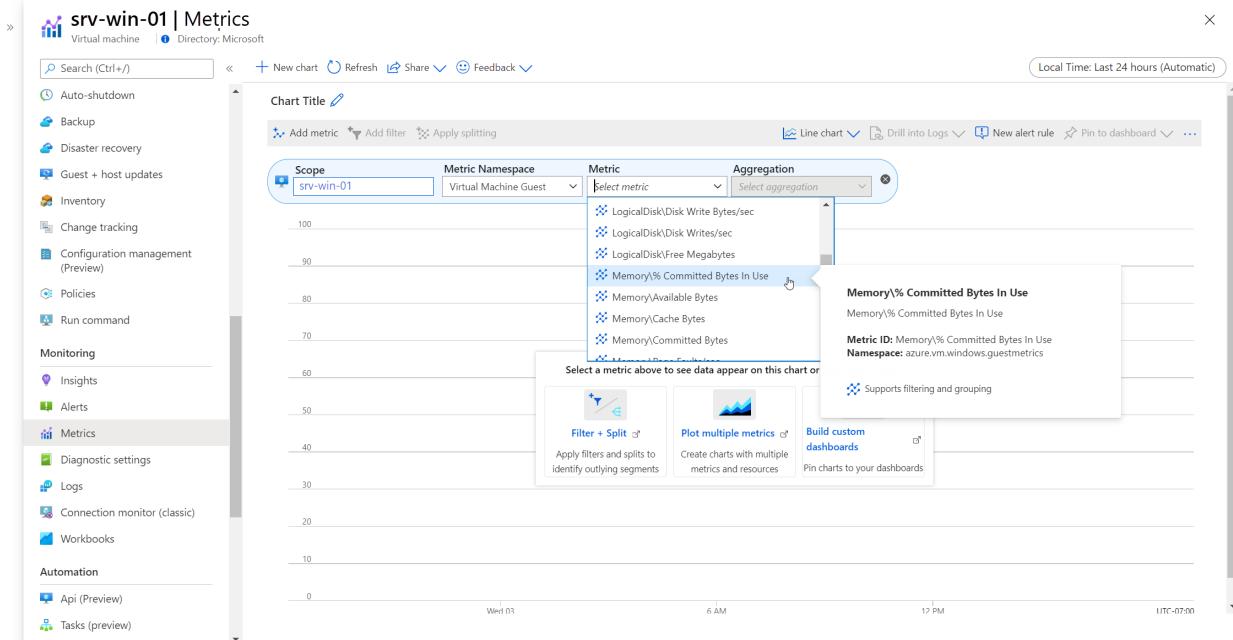
Select **Metrics** from your virtual machine's menu. Metrics explorer opens with the scope set to your virtual machine. Click **Metric Namespace**, and select **Virtual Machine Guest**.

### NOTE

If you don't see **Virtual Machine Guest**, you may just need to wait a few more minutes for the agent to be deployed and data to begin collecting.

The screenshot shows the Azure Metrics Explorer interface for the virtual machine 'srv-win-01'. The left sidebar has a 'Metrics' section selected. The main area shows a chart with a dropdown menu for 'Metric Namespace'. The dropdown menu has two options: 'Virtual Machine Host' (which is selected) and 'Virtual Machine Guest'. Below the chart are three buttons: 'Filter + Split', 'Plot multiple metrics', and 'Build custom dashboards'.

The available guest metrics are displayed. Select a **Metric** to add to the chart.



You can get a complete tutorial on viewing and analyzing metric data using metrics explorer in [Tutorial: Analyze metrics for an Azure resource](#) and on creating metrics alerts in [Tutorial: Create a metric alert for an Azure resource](#).

## Next steps

Now that you're collecting guest metrics for the virtual machine, you can create metric alerts based on guest metrics such as available memory and logical disk space.

[Create a metric alert in Azure Monitor](#)

# Azure Monitor Agent overview

9/21/2022 • 8 minutes to read • [Edit Online](#)

Azure Monitor Agent (AMA) collects monitoring data from the guest operating system of Azure and hybrid virtual machines and delivers it to Azure Monitor for use by features, insights, and other services, such as [Microsoft Sentinel](#) and [Microsoft Defender for Cloud](#). Azure Monitor Agent replaces all of Azure Monitor's legacy monitoring agents. This article provides an overview of Azure Monitor Agent's capabilities and supported use cases.

Here's a short [introduction to Azure Monitor video](#), which includes a quick demo of how to set up the agent from the Azure portal: [ITOps Talk: Azure Monitor Agent](#)

## Consolidating legacy agents

Deploy Azure Monitor Agent on all new virtual machines, scale sets and on-premises servers to collect data for [supported services and features](#).

If you have machines already deployed with legacy Log Analytics agents, we recommend you [migrate to Azure Monitor Agent](#) as soon as possible. The legacy Log Analytics agent will not be supported after August 2024.

Azure Monitor Agent replaces the Azure Monitor legacy monitoring agents:

- [Log Analytics Agent](#): Sends data to a Log Analytics workspace and supports monitoring solutions. This is fully consolidated into Azure Monitor agent.
- [Telegraf agent](#): Sends data to Azure Monitor Metrics (Linux only). Only basic Telegraf plugins are supported today in Azure Monitor agent.
- [Diagnostics extension](#): Sends data to Azure Monitor Metrics (Windows only), Azure Event Hubs, and Azure Storage. This is not consolidated yet.

## Install the agent and configure data collection

Azure Monitor Agent uses [data collection rules](#), using which you define which data you want each agent to collect. Data collection rules let you manage data collection settings at scale and define unique, scoped configurations for subsets of machines. The rules are independent of the workspace and the virtual machine, which means you can define a rule once and reuse it across machines and environments.

### To collect data using Azure Monitor Agent:

1. Install the agent on the resource.

RESOURCE TYPE	INSTALLATION METHOD	MORE INFORMATION
Virtual machines, scale sets	<a href="#">Virtual machine extension</a>	Installs the agent by using Azure extension framework.
On-premises servers (Azure Arc-enabled servers)	<a href="#">Virtual machine extension</a> (after installing the <a href="#">Azure Arc agent</a> )	Installs the agent by using Azure extension framework, provided for on-premises by first installing <a href="#">Azure Arc agent</a> .
Windows 10, 11 desktops, workstations	<a href="#">Client installer (Public preview)</a>	Installs the agent by using a Windows MSI installer.

RESOURCE TYPE	INSTALLATION METHOD	MORE INFORMATION
Windows 10, 11 laptops	<a href="#">Client installer (Public preview)</a>	Installs the agent by using a Windows MSI installer. The installer works on laptops, but the agent <i>isn't optimized yet</i> for battery or network consumption.

2. Define a data collection rule and associate the resource to the rule.

The table below lists the types of data you can currently collect with the Azure Monitor Agent and where you can send that data.

DATA SOURCE	DESTINATIONS	DESCRIPTION
Performance	Azure Monitor Metrics (Public preview) <sup>1</sup> - Insights.virtualmachine namespace Log Analytics workspace - <a href="#">Perf</a> table	Numerical values measuring performance of different aspects of operating system and workloads
Windows event logs (including sysmon events)	Log Analytics workspace - <a href="#">Event</a> table	Information sent to the Windows event logging system
Syslog	Log Analytics workspace - <a href="#">Syslog</a> <sup>2</sup> table	Information sent to the Linux event logging system
Text logs	Log Analytics workspace - custom table	Events sent to log file on agent machine

<sup>1</sup> On Linux, using Azure Monitor Metrics as the only destination is supported in v1.10.9.0 or higher.

<sup>2</sup> Azure Monitor Linux Agent v1.15.2 or higher supports syslog RFC formats including Cisco Meraki, Cisco ASA, Cisco FTD, Sophos XG, Juniper Networks, Corelight Zeek, CipherTrust, NXLog, McAfee, and Common Event Format (CEF).

## Supported services and features

In addition to the generally available data collection listed above, Azure Monitor Agent also supports these Azure Monitor features in preview:

AZURE MONITOR FEATURE	CURRENT SUPPORT	OTHER EXTENSIONS INSTALLED	MORE INFORMATION
Text logs and Windows IIS logs	Public preview	None	<a href="#">Collect text logs with Azure Monitor Agent (Public preview)</a>
Windows client installer	Public preview	None	<a href="#">Set up Azure Monitor Agent on Windows client devices</a>
<a href="#">VM insights</a>	Public preview	Dependency Agent extension, if you're using the Map Services feature	<a href="#">Enable VM Insights overview</a>

In addition to the generally available data collection listed above, Azure Monitor Agent also supports these Azure services in preview:

AZURE SERVICE	CURRENT SUPPORT	OTHER EXTENSIONS INSTALLED	MORE INFORMATION
Microsoft Defender for Cloud	Public preview	<ul style="list-style-type: none"> <li>Azure Security Agent extension</li> <li>SQL Advanced Threat Protection extension</li> <li>SQL Vulnerability Assessment extension</li> </ul>	<a href="#">Auto-deployment of Azure Monitor Agent (Preview)</a>
Microsoft Sentinel	<ul style="list-style-type: none"> <li>Windows Security Events: <a href="#">Generally available</a></li> <li>Windows Forwarding Event (WEF): <a href="#">Public preview</a></li> <li>Windows DNS logs: <a href="#">Public preview</a></li> <li>Linux Syslog CEF: Preview</li> </ul>	Sentinel DNS extension, if you're collecting DNS logs. For all other data types, you just need the Azure Monitor Agent extension.	<ul style="list-style-type: none"> <li><a href="#">Sign-up link for Linux Syslog CEF</a></li> <li>No sign-up needed for Windows Forwarding Event (WEF), Windows Security Events and Windows DNS events</li> </ul>
Change Tracking	Change Tracking: Preview.	Change Tracking extension	<a href="#">Sign-up link</a>
Update Management (available without Azure Monitor Agent)	Use Update Management v2 - Public preview	None	<a href="#">Update management center (Public preview) documentation</a>
Network Watcher	Connection Monitor: Preview	Azure NetworkWatcher extension	<a href="#">Sign-up link</a>

## Supported regions

Azure Monitor Agent is available in all public regions and Azure Government clouds. It's not yet supported in air-gapped clouds. For more information, see [Product availability by region](#).

## Costs

There's no cost for the Azure Monitor Agent, but you might incur charges for the data ingested. For information on Log Analytics data collection and retention and for customer metrics, see [Azure Monitor pricing](#).

## Compare to legacy agents

The tables below provide a comparison of Azure Monitor Agent with the legacy the Azure Monitor telemetry agents for Windows and Linux.

### Windows agents

		AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION (WAD)
Environments supported				

		AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION (WAD)
	Azure	X	X	X
	Other cloud (Azure Arc)	X	X	
	On-premises (Azure Arc)	X	X	
	Windows Client OS	X (Public preview)		
<b>Data collected</b>				
	Event Logs	X	X	X
	Performance	X	X	X
	File based logs	X (Public preview)	X	X
	IIS logs	X (Public preview)	X	X
	ETW events			X
	.NET app logs			X
	Crash dumps			X
	Agent diagnostics logs			X
<b>Data sent to</b>				
	Azure Monitor Logs	X	X	
	Azure Monitor Metrics <sup>1</sup>	X		X
	Azure Storage			X
	Event Hub			X
<b>Services and features supported</b>				
	Microsoft Sentinel	X ( <a href="#">View scope</a> )	X	
	VM Insights	X (Public preview)	X	
	Microsoft Defender for Cloud	X (Public preview)	X	

		AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION (WAD)
	Update Management	X (Public preview, independent of monitoring agents)	X	
	Change Tracking		X	

## Linux agents

		AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION (LAD)	TELEGRAF AGENT
<b>Environments supported</b>					
	Azure	X	X	X	X
	Other cloud (Azure Arc)	X	X		X
	On-premises (Azure Arc)	X	X		X
<b>Data collected</b>					
	Syslog	X	X	X	
	Performance	X	X	X	X
	File based logs	X (Public preview)			
<b>Data sent to</b>					
	Azure Monitor Logs	X	X		
	Azure Monitor Metrics <sup>1</sup>	X			X
	Azure Storage			X	
	Event Hub			X	
<b>Services and features supported</b>					
	Microsoft Sentinel	X ( <a href="#">View scope</a> )	X		
	VM Insights	X (Public preview)	X		

		AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION (LAD)	TELEGRAF AGENT
	Microsoft Defender for Cloud	X (Public preview)	X		
	Update Management	X (Public preview, independent of monitoring agents)	X		
	Change Tracking		X		

<sup>1</sup> To review other limitations of using Azure Monitor Metrics, see [quotas and limits](#). On Linux, using Azure Monitor Metrics as the only destination is supported in v.1.10.9.0 or higher.

## Supported operating systems

The following tables list the operating systems that Azure Monitor Agent and the legacy agents support. All operating systems are assumed to be x64. x86 isn't supported for any operating system.

### Windows

OPERATING SYSTEM	AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION
Windows Server 2022	X		
Windows Server 2022 Core	X		
Windows Server 2019	X	X	X
Windows Server 2019 Core	X		
Windows Server 2016	X	X	X
Windows Server 2016 Core	X		X
Windows Server 2012 R2	X	X	X
Windows Server 2012	X	X	X
Windows Server 2008 R2 SP1	X	X	X
Windows Server 2008 R2			X
Windows Server 2008 SP2		X	
Windows 11 Client Enterprise and Pro	X <sup>2, 3</sup>		
Windows 10 1803 (RS4) and higher	X <sup>2</sup>		

OPERATING SYSTEM	AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION
Windows 10 Enterprise (including multi-session) and Pro (Server scenarios only <sup>1</sup> )	X	X	X
Windows 8 Enterprise and Pro (Server scenarios only <sup>1</sup> )		X	
Windows 7 SP1 (Server scenarios only <sup>1</sup> )		X	
Azure Stack HCI		X	

<sup>1</sup> Running the OS on server hardware, for example, machines that are always connected, always turned on, and not running other workloads (PC, office, browser).

<sup>2</sup> Using the Azure Monitor agent [client installer \(Public preview\)](#).

<sup>3</sup> Also supported on Arm64-based machines.

#### Linux

OPERATING SYSTEM	AZURE MONITOR AGENT <sup>1</sup>	LOG ANALYTICS AGENT <sup>1</sup>	DIAGNOSTICS EXTENSION <sup>2</sup>
AlmaLinux 8.5	X <sup>3</sup>		
AlmaLinux 8	X	X	
Amazon Linux 2017.09		X	
Amazon Linux 2		X	
CentOS Linux 8	X	X	
CentOS Linux 7	X <sup>3</sup>	X	X
CentOS Linux 6		X	
CentOS Linux 6.5+		X	X
CBL-Mariner 2.0	X		
Debian 11	X <sup>3</sup>		
Debian 10	X	X	
Debian 9	X	X	X
Debian 8		X	
Debian 7			X
OpenSUSE 15	X		

OPERATING SYSTEM	AZURE MONITOR AGENT	LOG ANALYTICS AGENT	DIAGNOSTICS EXTENSION
OpenSUSE 13.1+			X
Oracle Linux 8	X	X	
Oracle Linux 7	X	X	X
Oracle Linux 6		X	
Oracle Linux 6.4+		X	X
Red Hat Enterprise Linux Server 8.6	X <sup>3</sup>		
Red Hat Enterprise Linux Server 8	X	X	
Red Hat Enterprise Linux Server 7	X	X	X
Red Hat Enterprise Linux Server 6		X	
Red Hat Enterprise Linux Server 6.7+		X	X
Rocky Linux 8	X	X	
SUSE Linux Enterprise Server 15 SP4	X <sup>3</sup>		
SUSE Linux Enterprise Server 15 SP2	X		
SUSE Linux Enterprise Server 15 SP1	X	X	
SUSE Linux Enterprise Server 15	X	X	
SUSE Linux Enterprise Server 12	X	X	X
Ubuntu 22.04 LTS	X		
Ubuntu 20.04 LTS	X <sup>3</sup>	X	X
Ubuntu 18.04 LTS	X <sup>3</sup>	X	X
Ubuntu 16.04 LTS	X	X	X
Ubuntu 14.04 LTS		X	X

<sup>1</sup> Requires Python (2 or 3) to be installed on the machine.

<sup>2</sup> Requires Python 2 to be installed on the machine and aliased to the `python` command.

<sup>3</sup> Also supported on Arm64-based machines.

## Next steps

- [Install the Azure Monitor Agent](#) on Windows and Linux virtual machines.
- [Create a data collection rule](#) to collect data from the agent and send it to Azure Monitor.



# Manage the Azure Monitor agent

9/21/2022 • 11 minutes to read • [Edit Online](#)

This article provides the different options currently available to install, uninstall, and update the [Azure Monitor agent](#). This agent extension can be installed on Azure virtual machines, scale sets, and Azure Arc-enabled servers. It also lists the options to create [associations with data collection rules](#) that define which data the agent should collect. Installing, upgrading, or uninstalling the Azure Monitor agent won't require you to restart your server.

## Virtual machine extension details

The Azure Monitor agent is implemented as an [Azure VM extension](#) with the details in the following table. You can install it by using any of the methods to install virtual machine extensions including the methods described in this article.

PROPERTY	WINDOWS	LINUX
Publisher	Microsoft.Azure.Monitor	Microsoft.Azure.Monitor
Type	AzureMonitorWindowsAgent	AzureMonitorLinuxAgent
TypeHandlerVersion	See <a href="#">Azure Monitor agent extension versions</a>	<a href="#">Azure Monitor agent extension versions</a>

## Extension versions

View [Azure Monitor agent extension versions](#).

## Prerequisites

The following prerequisites must be met prior to installing the Azure Monitor agent.

- **Permissions:** For methods other than using the Azure portal, you must have the following role assignments to install the agent:

BUILT-IN ROLE	SCOPES	REASON
• <a href="#">Virtual Machine Contributor</a> • <a href="#">Azure Connected Machine Resource Administrator</a>	• Virtual machines, scale sets, • Azure Arc-enabled servers	To deploy the agent
Any role that includes the action <code>Microsoft.Resources/deployments/*</code>	• Subscription and/or • Resource group and/or	To deploy Azure Resource Manager templates

- **Non-Azure:** To install the agent on physical servers and virtual machines hosted *outside* of Azure (that is, on-premises) or in other clouds, you must [install the Azure Arc Connected Machine agent](#) first, at no added cost.
- **Authentication:** [Managed identity](#) must be enabled on Azure virtual machines. Both user-assigned and

system-assigned managed identities are supported.

- **User-assigned:** This managed identity is recommended for large-scale deployments, configurable via [built-in Azure policies](#). You can create a user-assigned managed identity once and share it across multiple VMs, which means it's more scalable than a system-assigned managed identity. If you use a user-assigned managed identity, you must pass the managed identity details to the Azure Monitor agent via extension settings:

```
{  
  "authentication": {  
    "managedIdentity": {  
      "identifier-name": "mi_res_id" or "object_id" or "client_id",  
      "identifier-value": "<resource-id-of-uai>" or "<guid-object-or-client-id>"  
    }  
  }  
}
```

We recommend that you use `mi_res_id` as the `identifier-name`. The following sample commands only show usage with `mi_res_id` for the sake of brevity. For more information on `mi_res_id`, `object_id`, and `client_id`, see the [Managed identity documentation](#).

- **System-assigned:** This managed identity is suited for initial testing or small deployments. When used at scale, for example, for all VMs in a subscription, it results in a substantial number of identities created (and deleted) in Azure Active Directory. To avoid this churn of identities, use user-assigned managed identities instead. *For Azure Arc-enabled servers, system-assigned managed identity is enabled automatically as soon as you install the Azure Arc agent. It's the only supported type for Azure Arc-enabled servers.*
- **Not required for Azure Arc-enabled servers:** The system identity is enabled automatically if the agent is installed via [creating and assigning a data collection rule by using the Azure portal](#).
- **Networking:** If you use network firewalls, the [AzureResourceManager service tag](#) must be enabled on the virtual network for the virtual machine. The virtual machine must also have access to the following HTTPS endpoints:
  - `global.handler.control.monitor.azure.com`
  - `<virtual-machine-region-name>.handler.control.monitor.azure.com` (example: `westus.handler.control.azure.com`)
  - `<log-analytics-workspace-id>.ods.opinsights.azure.com` (example: `12345a01-b1cd-1234-e1f2-1234567g8h99.ods.opinsights.azure.com`)  
(If you use private links on the agent, you must also add the [dce endpoints](#)).

#### NOTE

This article only pertains to agent installation or management. After you install the agent, you must review the next article to [configure data collection rules and associate them with the machines](#) with agents installed. *The Azure Monitor agents can't function without being associated with data collection rules.*

## Use the Azure portal

Follow these instructions to use the Azure portal.

### Install

To install the Azure Monitor agent by using the Azure portal, follow the process to [create a data collection rule](#) in the Azure portal. This process creates the rule, associates it to the selected resources, and installs the Azure

Monitor agent on them if it's not already installed.

## Uninstall

To uninstall the Azure Monitor agent by using the Azure portal, go to your virtual machine, scale set, or Azure Arc-enabled server. Select the **Extensions** tab and select **AzureMonitorWindowsAgent** or **AzureMonitorLinuxAgent**. In the dialog that opens, select **Uninstall**.

## Update

To perform a one-time update of the agent, you must first uninstall the existing agent version. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature. Go to your virtual machine or scale set, select the **Extensions** tab and select **AzureMonitorWindowsAgent** or **AzureMonitorLinuxAgent**. In the dialog that opens, select **Enable automatic upgrade**.

# Use Resource Manager templates

Follow these instructions to use Azure Resource Manager templates.

## Install

You can use Resource Manager templates to install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers and to create an association with data collection rules. You must create any data collection rule prior to creating the association.

Get sample templates for installing the agent and creating the association from the following resources:

- [Template to install Azure Monitor agent \(Azure and Azure Arc\)](#)
- [Template to create association with data collection rule](#)

Install the templates by using [any deployment method for Resource Manager templates](#), such as the following commands.

- [PowerShell](#)
- [CLI](#)

```
New-AzResourceGroupDeployment -ResourceGroupName "<resource-group-name>" -TemplateFile "<template-filename.json>" -TemplateParameterFile "<parameter-filename.json>"
```

# Use PowerShell

You can install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers by using the PowerShell command for adding a virtual machine extension.

## Install on Azure virtual machines

Use the following PowerShell commands to install the Azure Monitor agent on Azure virtual machines. Choose the appropriate command based on your chosen authentication method.

### User-assigned managed identity

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Location <location> -TypeHandlerVersion <version-number> -SettingString '{"authentication":{"managedIdentity":{"identifier-name":"mi_res_id","identifier-value":/subscriptions/<my-subscription-id>/resourceGroups/<my-resource-group>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<my-user-assigned-identity>"}}}'
```

### System-assigned managed identity

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Location <location> -TypeHandlerVersion <version-number>
```

### Uninstall on Azure virtual machines

Use the following PowerShell commands to uninstall the Azure Monitor agent on Azure virtual machines.

- [Windows](#)
- [Linux](#)

```
Remove-AzVMExtension -Name AzureMonitorWindowsAgent -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name>
```

### Update on Azure virtual machines

To perform a one-time update of the agent, you must first uninstall the existing agent version,. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
Set-AzVMExtension -ExtensionName AzureMonitorWindowsAgent -ResourceGroupName <resource-group-name> -VMName <virtual-machine-name> -Publisher Microsoft.Azure.Monitor -ExtensionType AzureMonitorWindowsAgent -TypeHandlerVersion <version-number> -Location <location> -EnableAutomaticUpgrade $true
```

### Install on Azure Arc-enabled servers

Use the following PowerShell commands to install the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
New-AzConnectedMachineExtension -Name AzureMonitorWindowsAgent -ExtensionType AzureMonitorWindowsAgent -Publisher Microsoft.Azure.Monitor -ResourceGroupName <resource-group-name> -MachineName <arc-server-name> -Location <arc-server-location>
```

### Uninstall on Azure Arc-enabled servers

Use the following PowerShell commands to uninstall the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
Remove-AzConnectedMachineExtension -MachineName <arc-server-name> -ResourceGroupName <resource-group-name> -Name AzureMonitorWindowsAgent
```

## Upgrade on Azure Arc-enabled servers

To perform a one-time upgrade of the agent, use the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
$target = @{"Microsoft.Azure.Monitor.AzureMonitorWindowsAgent" = @{"targetVersion"=<target-version-number>}}
Update-AzConnectedExtension -ResourceGroupName $env.ResourceGroupName -MachineName <arc-server-name> -ExtensionTarget $target
```

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade \(preview\)](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
Update-AzConnectedMachineExtension -ResourceGroup <resource-group-name> -MachineName <arc-server-name> -Name AzureMonitorWindowsAgent -EnableAutomaticUpgrade
```

# Use the Azure CLI

You can install the Azure Monitor agent on Azure virtual machines and on Azure Arc-enabled servers by using the Azure CLI command for adding a virtual machine extension.

## Install on Azure virtual machines

Use the following CLI commands to install the Azure Monitor agent on Azure virtual machines. Choose the appropriate command based on your chosen authentication method.

### User-assigned managed identity

- [Windows](#)
- [Linux](#)

```
az vm extension set --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --ids <vm-resource-id> --settings '{"authentication":{"managedIdentity":{"identifier-name":"mi_res_id","identifier-value":/subscriptions/<my-subscription-id>/resourceGroups/<my-resource-group>/providers/Microsoft.ManagedIdentity/userAssignedIdentities/<my-user-assigned-identity>"}}}'
```

### System-assigned managed identity

- [Windows](#)
- [Linux](#)

```
az vm extension set --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --ids <vm-resource-id>
```

## Uninstall on Azure virtual machines

Use the following CLI commands to uninstall the Azure Monitor agent on Azure virtual machines.

- [Windows](#)
- [Linux](#)

```
az vm extension delete --resource-group <resource-group-name> --vm-name <virtual-machine-name> -name AzureMonitorWindowsAgent
```

## Update on Azure virtual machines

To perform a one-time update of the agent, you must first uninstall the existing agent version,. Then install the new version as described.

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade](#) feature by using the following CLI commands.

- [Windows](#)
- [Linux](#)

```
az vm extension set -name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --vm-name <virtual-machine-name> --resource-group <resource-group-name> --enable-auto-upgrade true
```

## Install on Azure Arc-enabled servers

Use the following CLI commands to install the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension create --name AzureMonitorWindowsAgent --publisher Microsoft.Azure.Monitor --type AzureMonitorWindowsAgent --machine-name <arc-server-name> --resource-group <resource-group-name> --location <arc-server-location>
```

## Uninstall on Azure Arc-enabled servers

Use the following CLI commands to uninstall the Azure Monitor agent on Azure Arc-enabled servers.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension delete --name AzureMonitorWindowsAgent --machine-name <arc-server-name> --resource-group <resource-group-name>
```

## Upgrade on Azure Arc-enabled servers

To perform a one-time upgrade of the agent, use the following CLI commands.

- [Windows](#)
- [Linux](#)

```
az connectedmachine upgrade-extension --extension-targets "  
{\"Microsoft.Azure.Monitor.AzureMonitorWindowsAgent\":{\"targetVersion\":\"<target-version-number>\"},\"}  
--machine-name <arc-server-name> --resource-group <resource-group-name>
```

We recommend that you enable automatic update of the agent by enabling the [Automatic Extension Upgrade \(preview\)](#) feature by using the following PowerShell commands.

- [Windows](#)
- [Linux](#)

```
az connectedmachine extension update --name AzureMonitorWindowsAgent --machine-name <arc-server-name> --  
resource-group <resource-group-name> --enable-auto-upgrade true
```

## Use Azure Policy

Use the following policies and policy initiatives to automatically install the agent and associate it with a data collection rule every time you create a virtual machine, scale set, or Azure Arc-enabled server.

### NOTE

As per Microsoft Identity best practices, policies for installing the Azure Monitor agent on virtual machines and scale sets rely on user-assigned managed identity. This option is the more scalable and resilient managed identity for these resources. For Azure Arc-enabled servers, policies rely on system-assigned managed identity as the only supported option today.

### Built-in policy initiatives

Before you proceed, review [prerequisites for agent installation](#).

Policy initiatives for Windows and Linux virtual machines, scale sets consist of individual policies that:

- (Optional) Create and assign built-in user-assigned managed identity, per subscription, per region. [Learn more](#).
  - **Bring Your Own User-Assigned Identity** : If set to `true`, it creates the built-in user-assigned managed identity in the predefined resource group and assigns it to all machines that the policy is applied to. If set to `false`, you can instead use existing user-assigned identity that *you must assign* to the machines beforehand.
- Install the Azure Monitor agent extension on the machine, and configure it to use user-assigned identity as specified by the following parameters.
  - **Bring Your Own User-Assigned Managed Identity** : If set to `false`, it configures the agent to use the built-in user-assigned managed identity created by the preceding policy. If set to `true`, it configures the agent to use an existing user-assigned identity that *you must assign* to the machines in scope beforehand.
  - **User-Assigned Managed Identity Name** : If you use your own identity (selected `true`), specify the name of the identity that's assigned to the machines.
  - **User-Assigned Managed Identity Resource Group** : If you use your own identity (selected `true`), specify the resource group where the identity exists.
  - **Additional Virtual Machine Images** : Pass additional VM image names that you want to apply the policy to, if not already included.
- Create and deploy the association to link the machine to specified data collection rule.

- **Data Collection Rule Resource Id** : The Azure Resource Manager resourceId of the rule you want to associate via this policy to all machines the policy is applied to.

Home > Policy >

**Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection Rule**

Initiative Definition

[Assign](#) [Edit initiative](#) [Duplicate initiative](#) [Delete initiative](#) [Export initiative](#)

^ Essentials

Name	: Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection ...	Definition location	: --
Description	: Monitor your Windows virtual machines and virtual machine scale sets by deploying the Azure Monitor Agent extension wit...	Definition ID	: /providers/Microsoft.Authorization/policySetDefinitions/0d1b56c6-6d11-4a5d-8
Category	: Monitoring	Type	: Built-in
Version	: 1.0.0		

Automated Microsoft managed Attestation Assignments (0) Parameters

Filter by reference ID, policy name... All effects All types

Policy ↑↓	Effect Type ↑↓	Type ↑↓	Reference ID ↑↓
[?] Assign Built-In User-Assigned Managed Identity to Virtual Machines	[parameters('effect')]	Built-in	addUserAssignedManagedIdentity
[?] Assign Built-In User-Assigned Managed Identity to Virtual Machine Scale Sets	[parameters('effect')]	Built-in	addUserAssignedManagedIdentity
[?] Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication	[parameters('effect')]	Built-in	deployAzureMonitoringAgent
[?] Configure Windows virtual machine scale sets to run Azure Monitor Agent with user-assigned managed identity-based authentication	[parameters('effect')]	Built-in	deployAzureMonitoringAgentWi
[?] Configure Windows Machines to be associated with a Data Collection Rule	[parameters('effect')]	Built-in	associateDataCollectionRuleV

## Known issues

- Managed Identity default behavior. [Learn more](#).
- Possible race condition with using built-in user-assigned identity creation policy. [Learn more](#).
- Assigning policy to resource groups. If the assignment scope of the policy is a resource group and not a subscription, the identity used by policy assignment (different from the user-assigned identity used by agent) must be manually granted [these roles](#) prior to assignment/remediation. Failing to do this step will result in *deployment failures*.
- Other [Managed Identity limitations](#).

## Built-in policies

You can choose to use the individual policies from the preceding policy initiative to perform a single action at scale. For example, if you *only* want to automatically install the agent, use the second agent installation policy from the initiative, as shown.

Home > Policy > Deploy Windows Azure Monitor Agent with user-assigned managed identity-based auth and associate with Data Collection Rule >

**Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication**

Policy definition

[Assign](#) [Edit definition](#) [Duplicate definition](#) [Delete definition](#) [Export definition](#)

^ Essentials

Name	: Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based auth...	Definition location	: --
Description	: Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telem...	Definition ID	: /providers/Microsoft.Authorization/policyDefinitions/637125fd-7c39-4b
Available Effects	: DeployIfNotExists, Disabled	Type	: Built-in
Category	: Monitoring	Mode	: Indexed

Definition Assignments (0) Parameters

```

1 {
2   "properties": {
3     "displayName": "Configure Windows virtual machines to run Azure Monitor Agent with user-assigned managed identity-based authentication",
4     "policyType": "BuiltIn",
5     "mode": "Indexed",
6     "description": "Automate the deployment of Azure Monitor Agent extension on your Windows virtual machines for collecting telemetry data from the guest OS. This policy is triggered when a new VM is created or updated. It deploys the Azure Monitor Agent extension to the VM and associates it with the Data Collection Rule defined in the parent policy initiative.",
7     "metadata": {
8       "version": "1.0.0",
9       "category": "Monitoring"
10    },
11    "parameters": {

```

## Remediation

The initiatives or policies will apply to each virtual machine as it's created. A [remediation task](#) deploys the policy definitions in the initiative to existing resources, so you can configure the Azure Monitor agent for any resources that were already created.

When you create the assignment by using the Azure portal, you have the option of creating a remediation task at the same time. For information on the remediation, see [Remediate non-compliant resources with Azure Policy](#).

## Configure Azure Monitor Agent to Linux virtual machines and associate to Data Collection Rule

Assign initiative

Basics Parameters **Remediation** Non-compliance messages Review + create

By default, this assignment will only take effect on newly created resources. Existing resources can be updated via a remediation task after the policy is assigned. For deployIfNotExists policies, the remediation task will deploy the specified template. For modify policies, the remediation task will edit tags on the existing resources.

Create a remediation task ⓘ

Policy to remediate

Configure Linux virtual machines with Azure Monitor Agent

### Managed Identity

Policies with the deployIfNotExists and modify effect types need the ability to deploy resources and edit tags on existing resources respectively. To do this, a managed identity will be created for you.

[Learn more about Managed Identity.](#)

Create a Managed Identity ⓘ

Managed identity location \*

East US

### Permissions

This identity will also be given the following permissions:

Virtual Machine Contributor, Monitoring Contributor

 Role assignments (permissions) are created based on the role definitions specified in the policies.

[Review + create](#)

[Cancel](#)

[Previous](#)

[Next](#)

## Next steps

[Create a data collection rule](#) to collect data from the agent and send it to Azure Monitor.

# Collect data from virtual machines with the Azure Monitor agent

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article describes how to collect events and performance counters from virtual machines using the Azure Monitor agent.

To collect data from virtual machines using the Azure Monitor agent, you'll:

1. Create [data collection rules \(DCR\)](#) that define which data Azure Monitor agent sends to which destinations.
2. Associate the data collection rule to specific virtual machines.

You can associate virtual machines to multiple data collection rules. This allows you to define each data collection rule to address a particular requirement, and associate the data collection rules to virtual machines based on the specific data you want to collect from each machine.

## Create data collection rule and association

To send data to Log Analytics, create the data collection rule in the **same region** as your Log Analytics workspace. You can still associate the rule to machines in other supported regions.

- [Portal](#)
- [API](#)
- [PowerShell](#)
- [Azure CLI](#)
- [Resource Manager template](#)

1. From the **Monitor** menu, select **Data Collection Rules**.
2. Select **Create** to create a new Data Collection Rule and associations.

The screenshot shows the Azure Monitor interface for Data Collection Rules. The left sidebar has a 'Data Collection Rules' section selected. The main area displays a table of existing rules:

Name	Subscription	Resource group	Location	Data sources	Destinations
central-it-default	my-subscription	my-resource-group	East US	Windows event logs	Azure Monitor Logs
lob-app	my-subscription	my-resource-group	East US	Performance counters	Azure Monitor Metrics
sql	my-subscription	my-resource-group	East US	Linux syslog	Azure Monitor Logs

3. Provide a **Rule name** and specify a **Subscription**, **Resource Group**, **Region**, and **Platform Type**.

**Region** specifies where the DCR will be created. The virtual machines and their associations can be in any subscription or resource group in the tenant.

**Platform Type** specifies the type of resources this rule can apply to. Custom allows for both Windows and Linux types.

Home > Monitor >

## Create Data Collection Rule

Data collection rule management

**Basics**   Resources   Collect and deliver   Review + create

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all of your resources.

**Rule details**

Rule Name \*  ✓

Subscription \*  ▼

Resource Group \*  ▼

Create new

Region \*  ▼

Platform Type \*  Windows  
 Linux  
 Custom

**Review + create**   < Previous   **Next : Resources >**

- On the **Resources** tab, add the resources (virtual machines, virtual machine scale sets, Arc for servers) to which to associate the data collection rule. The portal will install Azure Monitor Agent on resources that don't already have it installed, and will also enable Azure Managed Identity.

### IMPORTANT

The portal enables System-Assigned managed identity on the target resources, in addition to existing User-Assigned Identities (if any). For existing applications, unless you specify the User-Assigned identity in the request, the machine will default to using System-Assigned Identity instead.

If you need network isolation using private links, select existing endpoints from the same region for the respective resources, or [create a new endpoint](#).

Home > Monitor >

## Create Data Collection Rule

Data collection rule management

**Basics**   **Resources**   Collect and deliver   Review + create

Pick a set of machines to collect data from. The Azure Monitor Agent will be automatically installed on these machines.

This will also enable System Assigned Managed Identity on these machines, in addition to existing User Assigned Identities (if any). Note: Unless specified in the request, the machine will default to using System Assigned Identity for all other apps. [Learn More](#)

**+ Add Resource(s)**

⚠ Currently, only resources in the following region(s) are supported: Australia Southeast, Canada Central, Japan East, Australia East, Central India, Germany West Central, North Central US, South Central US, East US, Central US, West Europe, West US 2, Southeast Asia East US 2, UK South, North Europe, West US, Australia Central, West Central US, East Asia, UK West, Korea Central, France Central, South Africa North, Switzerland North, Brazil South, Australia Central 2, Brazil Southeast, Canada East, France South, Korea South, Norway West, UAE North, Japan West, Norway East, Switzerland West, East US 2, EUAP.

Name	Type	Location	Resource group	Subscription
my-vm-01	microsoft.compute/virtualmachines	westus	my-resource-grp	My subscription
my-vm-02	microsoft.compute/virtualmachines	westus	my-resource-grp	My subscription
my-vmss-01	microsoft.compute/virtualmachinescalesets	eastus	my-resource-grp	My subscription

**Review + create**   < Previous   **Next : Collect and deliver >**

- On the **Collect and deliver** tab, select **Add data source** to add a data source and set a destination.
- Select a **Data source type**.
- Select which data you want to collect. For performance counters, you can select from a predefined set of objects and their sampling rate. For events, you can select from a set of logs and severity levels.

Home > Monitor | Data Collection Rules >

## Create Data Collection Rule

Data collection rule management

**\*Basics**   Resources   **Collect and deliver**   Review + create

Configure which data sources to collect, and where to send the data to.

**+ Add data source**

No data sources and destination added

✖ This data collection rule doesn't have any data sources or destinations selected.

**Add data source**

**\* Data source**   Destination

Select which data source type and the data to collect for your virtual machine(s).

Data source type  ▼

**Configure Performance Counters**

Choose Basic to enable the collection of performance counters. Choose Custom if you want more control over which performance counters are collected.

None	Basic	Custom
<input checked="" type="checkbox"/> Performance counter	Sample rate (seconds)	
<input checked="" type="checkbox"/> CPU	10	
<input checked="" type="checkbox"/> Memory	10	
<input checked="" type="checkbox"/> Disk	10	
<input checked="" type="checkbox"/> Network	10	

**Review + create**   < Previous   **Next : Review + create >**

- Select **Custom** to collect logs and performance counters that are not [currently supported data sources](#) or to [filter events using XPath queries](#). You can then specify an [XPath](#) to collect any specific values. See [Sample DCR](#) for an example.

The screenshot shows the 'Add data source' dialog box. On the left, the 'Create Data Collection Rule' page is visible, showing the 'Collect and deliver' tab selected. On the right, the 'Add data source' dialog has the 'Data source' tab selected. Under 'Configure Performance Counters', there are tabs for 'None', 'Basic', and 'Custom'. The 'Custom' tab is selected, showing a list of performance counters with their sample rates:

Performance counter	Sample rate (seconds)
\Processor Information(_Total)\% Processor Time	10
\Processor Information(_Total)\% Privileged Time	10
\Processor Information(_Total)\% User Time	10
\Processor Information(_Total)\Processor Frequency	10
\System\Processes	10
\Process(_Total)\Thread Count	10
\Process(_Total)\Handle Count	10
\System\System Up Time	10
\System\Context Switches/sec	10
\System\Processor Queue Length	10

- On the **Destination** tab, add one or more destinations for the data source. You can select multiple destinations of the same or different types - for instance multiple Log Analytics workspaces (known as "multi-homing").

You can send Windows event and Syslog data sources to Azure Monitor Logs only. You can send performance counters to both Azure Monitor Metrics and Azure Monitor Logs.

The screenshot shows the 'Add data source' dialog box. On the left, the 'Create Data Collection Rule' page is visible, showing the 'Collect and deliver' tab selected. On the right, the 'Add data source' dialog has the 'Destination' tab selected. It shows two destination entries:

Destination type	Subscription	Account or namespace
Azure Monitor Metrics	my-subscription	No need to make a selection
Azure Monitor Logs	my-subscription	my-workspace

- Select **Add Data Source** and then **Review + create** to review the details of the data collection rule and association with the set of virtual machines.
- Select **Create** to create the data collection rule.

#### NOTE

It might take up to 5 minutes for data to be sent to the destinations after you create the data collection rule and associations.

## Filter events using XPath queries

Since you're charged for any data you collect in a Log Analytics workspace, collect only the data you need. The basic configuration in the Azure portal provides you with a limited ability to filter out events.

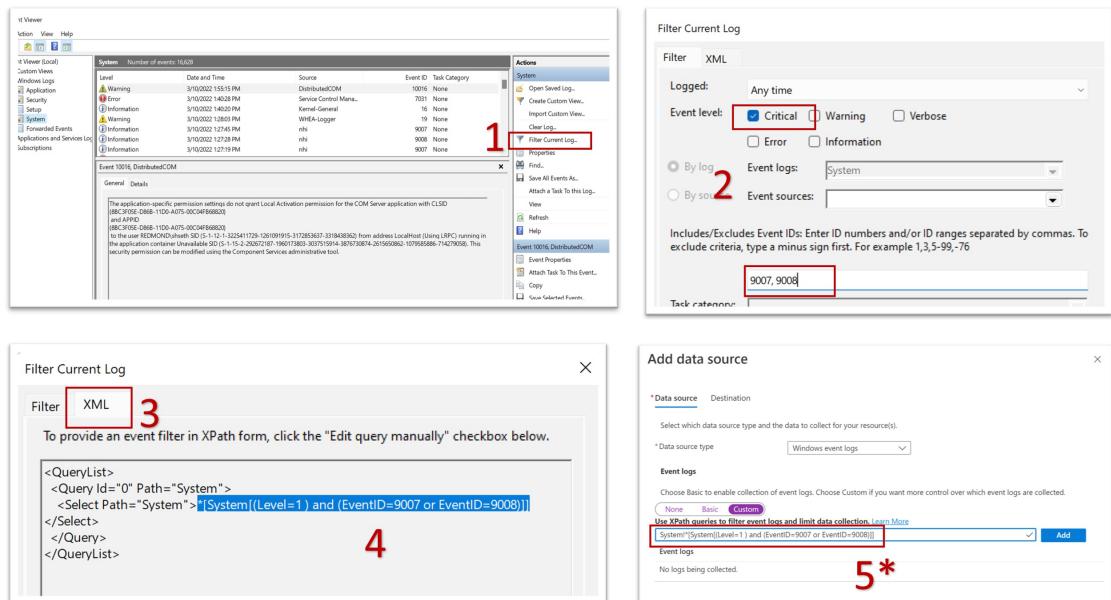
To specify additional filters, use Custom configuration and specify an XPath that filters out the events you don't need. XPath entries are written in the form `LogName!XPathQuery`. For example, you may want to return only events from the Application event log with an event ID of 1035. The XPathQuery for these events would be

`*[System[EventID=1035]]`. Since you want to retrieve the events from the Application event log, the XPath is `Application!*[System[EventID=1035]]`

## Extracting XPath queries from Windows Event Viewer

In Windows, you can use Event Viewer to extract XPath queries as shown below.

When you paste the XPath query into the field on the **Add data source** screen, (step 5 in the picture below), you must append the log type category followed by '!'.



See [XPath 1.0 limitations](#) for a list of limitations in the XPath supported by Windows event log.

### TIP

You can use the PowerShell cmdlet `Get-WinEvent` with the `FilterXPath` parameter to test the validity of an XPathQuery locally on your machine first. The following script shows an example.

```
$XPath = '*[System[EventID=1035]]'
Get-WinEvent -LogName 'Application' -FilterXPath $XPath
```

- In the cmdlet above, the value of the `-LogName` parameter is the initial part of the XPath query until the '!'. The rest of the XPath query goes into the `$XPath` parameter.
- If the script returns events, the query is valid.
- If you receive the message *No events were found that match the specified selection criteria.*, the query may be valid, but there are no matching events on the local machine.
- If you receive the message *The specified query is invalid*, the query syntax is invalid.

Examples of filtering events using a custom XPath:

DESCRIPTION	XPATH
Collect only System events with Event ID = 4648	<code>System!*[System[EventID=4648]]</code>
Collect Security Log events with Event ID = 4648 and a process name of consent.exe	<code>Security!*[System[(EventID=4648)]] and *[EventData[@Name='ProcessName']='C:\Windows\System32\consent.e</code>
Collect all Critical, Error, Warning, and Information events from the System event log except for Event ID = 6 (Driver loaded)	<code>System!*[System[(Level=1 or Level=2 or Level=3) and (EventID != 6)]]</code>

DESCRIPTION	XPATH
Collect all success and failure Security events except for Event ID 4624 (Successful logon)	<pre>Security!* [System[(band(Keywords,13510798882111488)) and (EventID != 4624)]]</pre>

## Next steps

- [Collect text logs using Azure Monitor agent.](#)
- Learn more about the [Azure Monitor Agent](#).
- Learn more about [data collection rules](#).

# Migrate to Azure Monitor Agent from Log Analytics agent

9/21/2022 • 5 minutes to read • [Edit Online](#)

Azure Monitor Agent (AMA) replaces the Log Analytics agent (also known as MMA and OMS) for both Windows and Linux machines, in Azure and on premises. It introduces a simplified, flexible method of configuring collection configuration called [Data Collection Rules \(DCRs\)](#). This article outlines the benefits of migrating to Azure Monitor Agent (AMA) and provides guidance on how to implement a successful migration.

## IMPORTANT

The Log Analytics agent will be [retired on 31 August, 2024](#). If you are currently using the Log Analytics agent with Azure Monitor or other supported features and services, you should start planning your migration to Azure Monitor Agent using the information in this article.

## Benefits

Azure Monitor Agent provides the following benefits over legacy agents:

- **Security and performance**
  - Enhanced security through Managed Identity and Azure Active Directory (Azure AD) tokens (for clients).
  - A higher events per second (EPS) upload rate.
- **Cost savings** using data collection [using Data Collection Rules](#). Using Data Collection Rules is one of the most useful advantages of using Azure Monitor Agent.
  - DCRs lets you configure data collection for specific machines connected to a workspace as compared to the "all or nothing" approach of legacy agents.
  - Using DCRs you can define which data to ingest and which data to filter out to reduce workspace clutter and save on costs.
- **Simpler management** of data collection, including ease of troubleshooting
  - Easy [multihoming](#) on Windows and Linux.
  - Centralized, 'in the cloud' agent configuration makes every action simpler and more easily scalable throughout the data collection lifecycle, from onboarding to deployment to updates and changes over time.
  - Greater transparency and control of more capabilities and services, such as Sentinel, Defender for Cloud, and VM Insights.
- **A single agent** that consolidates all features necessary to address all telemetry data collection needs across servers and client devices (running Windows 10, 11). This is the goal, though Azure Monitor Agent currently converges with the Log Analytics agents.

## Migration plan considerations

Your migration plan to the Azure Monitor Agent should take into account:

- **Current and new feature requirements:** Review [Azure Monitor Agent's supported services and features](#) to ensure that Azure Monitor Agent has the features you require. If you currently use unsupported features you can temporarily do without, consider migrating to the new agent to benefit

from added security and reduced cost immediately. Use the [AMA Migration Helper](#) to discover what solutions and features you're using today that depend on the legacy agent.

If you use Microsoft Sentinel, see [Gap analysis for Microsoft Sentinel](#) for a comparison of the extra data collected by Microsoft Sentinel.

- **Installing Azure Monitor Agent alongside a legacy agent:** If you're setting up a **new environment** with resources, such as deployment scripts and onboarding templates, and you still need a legacy agent, assess the effort of migrating to Azure Monitor Agent later. If the setup will take a significant amount of rework, install Azure Monitor Agent together with a legacy agent in your new environment to decrease the migration effort.

Azure Monitor Agent can run alongside the legacy Log Analytics agents on the same machine so that you can continue to use existing functionality during evaluation or migration. While this allows you to begin the transition, ensure you understand the limitations:

- Be careful in collecting duplicate data from the same machine, which could skew query results and affect downstream features like alerts, dashboards or workbooks. For example, VM Insights uses the Log Analytics agent to send performance data to a Log Analytics workspace. You might also have configured the workspace to collect Windows events and Syslog events from agents. If you install Azure Monitor Agent and create a data collection rule for these events and performance data, you'll collect duplicate data. If you're using both agents to collect the same type of data, make sure the agents are **collecting data from different machines or sending the data to different destinations**. Collecting duplicate data also generates more charges for data ingestion and retention.
- Running two telemetry agents on the same machine consumes double the resources, including, but not limited to CPU, memory, storage space, and network bandwidth.

## Prerequisites

Review the [prerequisites](#) for use Azure Monitor Agent. For on-premises servers or other cloud managed servers, [installing the Azure Arc agent](#) is an important prerequisite that then helps to install the agent extension and other required extensions. Using Arc for this purpose comes at no added cost, and it's not mandatory to use Arc for server management overall (i.e. you can continue using your existing on-premises management solutions). Once Arc agent is installed, you can follow the same guidance below across Azure and on-premise for migration.

## Migration testing

To ensure safe deployment during migration, begin testing with few resources running Azure Monitor Agent in your nonproduction environment. After you validate the data collected on these test resources, roll out to production by following the same steps.

See [create new data collection rules](#) to start collecting some of the existing data types. Alternatively you can use the [DCR Config Generator](#) to convert existing legacy agent configuration into data collection rules. After you **validate** that data is flowing as expected with Azure Monitor Agent, check the **Category** column in the [Heartbeat](#) table for the value *Azure Monitor Agent* for AMA collected data. Ensure it matches data flowing through the existing Log Analytics agent.

## At-scale migration using Azure Policy

We recommend using [Azure Policy](#) to migrate a large number of agents. Start by analyzing your current monitoring setup with the Log Analytics agent using the [AMA Migration Helper](#) to find sources, such as virtual machines, virtual machine scale sets, and on-premises servers.

Use the [DCR Config Generator](#) to migrate legacy agent configuration, including data sources and destinations, from the workspace to the new DCRs.

**IMPORTANT**

Before you deploy a large number of agents, consider [configuring the workspace](#) to disable data collection for the Log Analytics agent. If you leave data collection for the Log Analytics agent enabled, you may collect duplicate data and increase your costs. You might choose to collect duplicate data for a short period during migration until you verify that you've deployed and configured Azure Monitor Agent correctly.

Validate that Azure Monitor Agent is collecting data as expected and all downstream dependencies, such as dashboards, alerts, and workbooks, function properly.

After you confirm that Azure Monitor Agent is collecting data properly, [uninstall the Log Analytics agent](#) from monitored resources. Clean up any configuration files, workspace keys, or certificates that were used previously by the Log Analytics agent.

**IMPORTANT**

Don't uninstall the legacy agent if you need to use it for System Center Operations Manager scenarios or others solutions not yet available on Azure Monitor Agent.

## Next steps

For more information, see:

- [Azure Monitor Agent overview](#)
- [Azure Monitor Agent migration for Microsoft Sentinel](#)
- [Frequently asked questions for Azure Monitor Agent migration](#)



# Log Analytics agent overview

9/21/2022 • 7 minutes to read • [Edit Online](#)

This article provides a detailed overview of the Log Analytics agent and the agent's system and network requirements and deployment methods.

## IMPORTANT

The Log Analytics agent is on a **deprecation path** and won't be supported after **August 31, 2024**. If you use the Log Analytics agent to ingest data to Azure Monitor, [migrate to the new Azure Monitor agent](#) prior to that date.

You might also see the Log Analytics agent referred to as Microsoft Monitoring Agent (MMA).

## Primary scenarios

Use the Log Analytics agent if you need to:

- Collect logs and performance data from Azure virtual machines or hybrid machines hosted outside of Azure.
- Send data to a Log Analytics workspace to take advantage of features supported by [Azure Monitor Logs](#), such as [log queries](#).
- Use [VM insights](#), which allows you to monitor your machines at scale and monitor their processes and dependencies on other resources and external processes.
- Manage the security of your machines by using [Microsoft Defender for Cloud](#) or [Microsoft Sentinel](#).
- Use [Azure Automation Update Management](#), [Azure Automation State Configuration](#), or [Azure Automation Change Tracking and Inventory](#) to deliver comprehensive management of your Azure and non-Azure machines.
- Use different [solutions](#) to monitor a particular service or application.

Limitations of the Log Analytics agent:

- Can't send data to Azure Monitor Metrics, Azure Storage, or Azure Event Hubs.
- Difficult to configure unique monitoring definitions for individual agents.
- Difficult to manage at scale because each virtual machine has a unique configuration.

## Comparison to other agents

For a comparison between the Log Analytics and other agents in Azure Monitor, see [Overview of Azure Monitor agents](#).

## Supported operating systems

For a list of the Windows and Linux operating system versions that are supported by the Log Analytics agent, see [Supported operating systems](#).

## Installation options

This section explains how to install the Log Analytics agent on different types of virtual machines and connect the machines to Azure Monitor.

**IMPORTANT**

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

**NOTE**

Cloning a machine with the Log Analytics Agent already configured is *not* supported. If the agent is already associated with a workspace, cloning won't work for "golden images."

**Azure virtual machine**

- Use [VM insights](#) to install the agent for a [single machine using the Azure portal](#) or for [multiple machines at scale](#). This installs the Log Analytics agent and [Dependency agent](#).
- Log Analytics VM extension for [Windows](#) or [Linux](#) can be installed with the Azure portal, Azure CLI, Azure PowerShell, or an Azure Resource Manager template.
- [Microsoft Defender for Cloud](#) can provision the Log Analytics agent on all supported Azure VMs and any new ones that are created if you enable it to monitor for security vulnerabilities and threats.
- Install for individual Azure virtual machines [manually from the Azure portal](#).
- Connect the machine to a workspace from the [Virtual machines](#) option in the [Log Analytics workspaces](#) menu in the Azure portal.

**Windows virtual machine on-premises or in another cloud**

- Use [Azure Arc-enabled servers](#) to deploy and manage the Log Analytics VM extension. Review the [deployment options](#) to understand the different deployment methods available for the extension on machines registered with Azure Arc-enabled servers.
- [Manually install](#) the agent from the command line.
- Automate the installation with [Azure Automation DSC](#).
- Use a [Resource Manager template with Azure Stack](#).

**Linux virtual machine on-premises or in another cloud**

- Use [Azure Arc-enabled servers](#) to deploy and manage the Log Analytics VM extension. Review the [deployment options](#) to understand the different deployment methods available for the extension on machines registered with Azure Arc-enabled servers.
- [Manually install](#) the agent calling a wrapper-script hosted on GitHub.
- Integrate [System Center Operations Manager](#) with Azure Monitor to forward collected data from Windows computers reporting to a management group.

## Data collected

The following table lists the types of data you can configure a Log Analytics workspace to collect from all connected agents. For a list of insights and solutions that use the Log Analytics agent to collect other kinds of data, see [What is monitored by Azure Monitor?](#).

DATA SOURCE	DESCRIPTION
<a href="#">Windows Event logs</a>	Information sent to the Windows event logging system
<a href="#">Syslog</a>	Information sent to the Linux event logging system

DATA SOURCE	DESCRIPTION
Performance	Numerical values measuring performance of different aspects of operating system and workloads
IIS logs	Usage information for IIS websites running on the guest operating system
Custom logs	Events from text files on both Windows and Linux computers

## Other services

The agent for Linux and Windows isn't only for connecting to Azure Monitor. Other services such as Microsoft Defender for Cloud and Microsoft Sentinel rely on the agent and its connected Log Analytics workspace. The agent also supports Azure Automation to host the Hybrid Runbook Worker role and other services such as [Change Tracking](#), [Update Management](#), and [Microsoft Defender for Cloud](#). For more information about the Hybrid Runbook Worker role, see [Azure Automation Hybrid Runbook Worker](#).

## Workspace and management group limitations

For details on connecting an agent to an Operations Manager management group, see [Configure agent to report to an Operations Manager management group](#).

- Windows agents can connect to up to four workspaces, even if they're connected to a System Center Operations Manager management group.
- The Linux agent doesn't support multi-homing and can only connect to a single workspace or management group.

## Security limitations

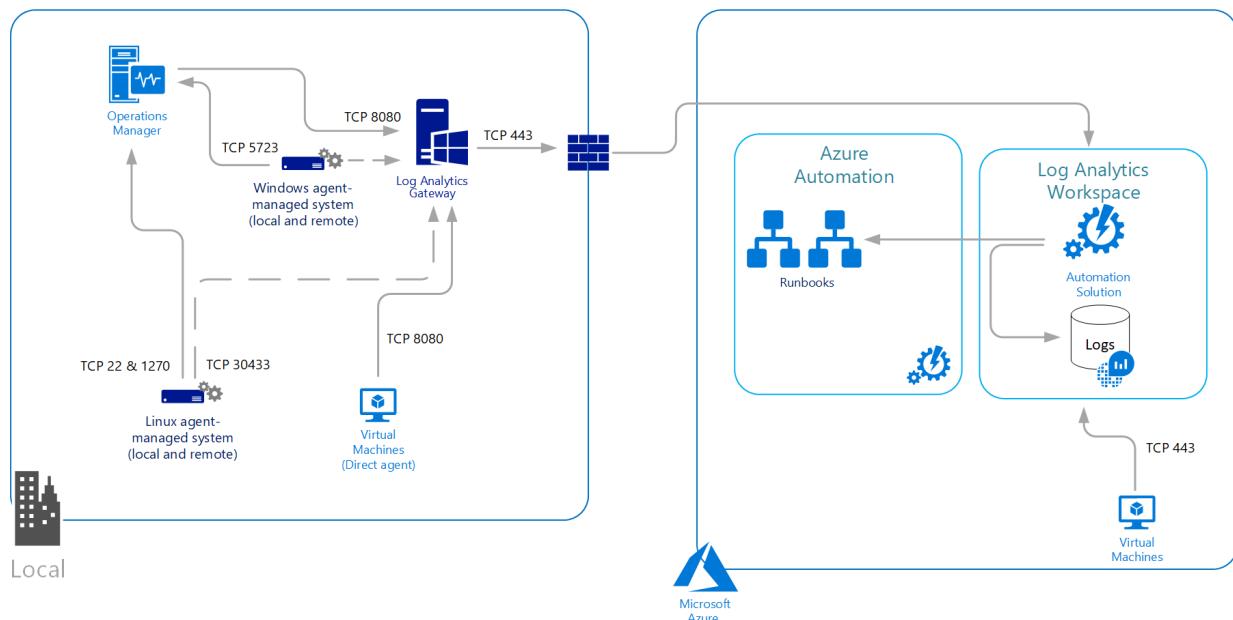
The Windows and Linux agents support the [FIPS 140 standard](#), but [other types of hardening might not be supported](#).

## TLS 1.2 protocol

To ensure the security of data in transit to Azure Monitor logs, we strongly encourage you to configure the agent to use at least Transport Layer Security (TLS) 1.2. Older versions of TLS/Secure Sockets Layer (SSL) have been found to be vulnerable. Although they still currently work to allow backward compatibility, they are *not recommended*. For more information, see [Sending data securely using TLS 1.2](#).

## Network requirements

The agent for Linux and Windows communicates outbound to the Azure Monitor service over TCP port 443. If the machine connects through a firewall or proxy server to communicate over the internet, review the following requirements to understand the network configuration required. If your IT security policies do not allow computers on the network to connect to the internet, set up a [Log Analytics gateway](#) and configure the agent to connect through the gateway to Azure Monitor. The agent can then receive configuration information and send data collected.



The following table lists the proxy and firewall configuration information required for the Linux and Windows agents to communicate with Azure Monitor logs.

### Firewall requirements

AGENT RESOURCE	POR TS	DIRECTION	BYPASS HTTPS INSPECTION
*.ods.opinsights.azure.com	Port 443	Outbound	Yes
*.oms.opinsights.azure.com	Port 443	Outbound	Yes
*.blob.core.windows.net	Port 443	Outbound	Yes
*.azure-automation.net	Port 443	Outbound	Yes

For firewall information required for Azure Government, see [Azure Government management](#).

#### IMPORTANT

If your firewall is doing CNAME inspections, you need to configure it to allow all domains in the CNAME.

If you plan to use the Azure Automation Hybrid Runbook Worker to connect to and register with the Automation service to use runbooks or management features in your environment, it must have access to the port number and the URLs described in [Configure your network for the Hybrid Runbook Worker](#).

### Proxy configuration

The Windows and Linux agent supports communicating either through a proxy server or Log Analytics gateway to Azure Monitor by using the HTTPS protocol. Both anonymous and basic authentication (username/password) are supported.

For the Windows agent connected directly to the service, the proxy configuration is specified during installation or [after deployment](#) from Control Panel or with PowerShell. Log Analytics Agent (MMA) doesn't use the system proxy settings. As a result, the user has to pass the proxy setting while installing MMA. These settings will be stored under MMA configuration (registry) on the virtual machine.

For the Linux agent, the proxy server is specified during installation or [after installation](#) by modifying the proxy.conf configuration file. The Linux agent proxy configuration value has the following syntax:

```
[protocol://][user:password@]proxyhost[:port]
```

PROPERTY	DESCRIPTION
Protocol	https
user	Optional username for proxy authentication
password	Optional password for proxy authentication
proxyhost	Address or FQDN of the proxy server/Log Analytics gateway
port	Optional port number for the proxy server/Log Analytics gateway

For example: `https://user01:password@proxy01.contoso.com:30443`

#### NOTE

If you use special characters such as "@" in your password, you'll receive a proxy connection error because the value is parsed incorrectly. To work around this issue, encode the password in the URL by using a tool like [URLDecode](#).

## Next steps

- Review [data sources](#) to understand the data sources available to collect data from your Windows or Linux system.
- Learn about [log queries](#) to analyze the data collected from data sources and solutions.
- Learn about [monitoring solutions](#) that add functionality to Azure Monitor and also collect data into the Log Analytics workspace.

# Install Log Analytics agent on Linux computers

9/21/2022 • 10 minutes to read • [Edit Online](#)

This article provides details on installing the Log Analytics agent on Linux computers hosted in other clouds or on-premises.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

The [installation methods described in this article](#) are:

- Install the agent for Linux using a wrapper-script hosted on GitHub. This is the recommended method to install and upgrade the agent when the computer has connectivity with the Internet, directly or through a proxy server.
- Manually download and install the agent. This is required when the Linux computer doesn't have access to the Internet and will be communicating with Azure Monitor or Azure Automation through the [Log Analytics gateway](#).

See [Installation options](#) for more efficient options you can use for Azure virtual machines.

## Requirements

### Supported operating systems

See [Overview of Azure Monitor agents](#) for a list of Linux distributions supported by the Log Analytics agent.

## NOTE

OpenSSL 1.1.0 is only supported on x86\_x64 platforms (64-bit) and OpenSSL earlier than 1.x is not supported on any platform.

## NOTE

The Log Analytics Linux Agent does not run in containers. To monitor containers, use the [Container Monitoring solution](#) for Docker hosts or [Container insights](#) for Kubernetes.

Starting with versions released after August 2018, we're making the following changes to our support model:

- Only the server versions are supported, not client.
- Focus support on any of the [Azure Linux Endorsed distros](#). There may be some delay between a new distro/version being Azure Linux Endorsed and it being supported for the Log Analytics Linux agent.
- All minor releases are supported for each major version listed.
- Versions that have passed their manufacturer's end-of-support date aren't supported.
- Only support VM images; containers, even those derived from official distro publishers' images, aren't supported.
- New versions of AMI aren't supported.
- Only versions that run OpenSSL 1.x by default are supported.

## NOTE

If you are using a distro or version that is not currently supported and doesn't align to our support model, we recommend that you fork this repo, acknowledging that Microsoft support will not provide assistance with forked agent versions.

## Python requirement

Starting from Agent version 1.13.27, the Linux Agent will support both Python 2 and 3. We always recommend using the latest agent.

If you're using an older version of the agent, you must have the Virtual Machine use Python 2 by default. If your virtual machine is using a distro that doesn't include Python 2 by default, then you must install it. The following sample commands will install Python 2 on different distros.

- Red Hat, CentOS, Oracle: `yum install -y python2`
- Ubuntu, Debian: `apt-get install -y python2`
- SUSE: `zypper install -y python2`

Again, only if you're using an older version of the agent, the `python2` executable must be aliased to `python`.

Following is one method that you can use to set this alias:

1. Run the following command to remove any existing aliases.

```
sudo update-alternatives --remove-all python
```

2. Run the following command to create the alias.

```
sudo update-alternatives --install /usr/bin/python python /usr/bin/python2 1
```

## Supported Linux hardening

The OMS Agent has limited customization and hardening support for Linux.

The following are currently supported:

- FIPS
- SELinux (Marketplace images for CentOS and RHEL with their default settings)

The following aren't supported:

- CIS
- SELinux (custom hardening like MLS)

CIS and SELinux hardening support is planned for [Azure Monitoring Agent](#). Further hardening and customization methods aren't supported nor planned for OMS Agent. For instance, OS images like GitHub Enterprise Server which include customizations such as limitations to user account privileges aren't supported.

## Agent prerequisites

The following table highlights the packages required for [supported Linux distros](#) that the agent will be installed on.

REQUIRED PACKAGE	DESCRIPTION	MINIMUM VERSION
Glibc	GNU C Library	2.5-12

REQUIRED PACKAGE	DESCRIPTION	MINIMUM VERSION
Openssl	OpenSSL Libraries	1.0.x or 1.1.x
Curl	cURL web client	7.15.5
Python		2.7 or 3.6+
Python-ctypes		
PAM	Pluggable Authentication Modules	

#### NOTE

Either rsyslog or syslog-ng are required to collect syslog messages. The default syslog daemon on version 5 of Red Hat Enterprise Linux, CentOS, and Oracle Linux version (sysklog) is not supported for syslog event collection. To collect syslog data from this version of these distributions, the rsyslog daemon should be installed and configured to replace sysklog.

## Network requirements

See [Log Analytics agent overview](#) for the network requirements for the Linux agent.

### Workspace ID and key

Regardless of the installation method used, you'll require the workspace ID and key for the Log Analytics workspace that the agent will connect to. Select the workspace from the **Log Analytics workspaces** menu in the Azure portal. Then select **Agents management** in the **Settings** section.

The screenshot shows the Azure Log Analytics workspace settings interface. On the left, there's a sidebar with options like 'Add', 'Open recycle bin', and 'Agents management'. The main area is titled 'CH-LA | Agents management' and shows '52 Windows computers connected'. It has sections for 'Download agent' (with links to 'Download Windows Agent (64 bit)' and 'Download Windows Agent (32 bit)'), 'Workspace ID' (a text input field containing '<GUID>'), 'Primary key' (a text input field containing '<GUID>'), and 'Secondary key' (a text input field containing '<GUID>'). Below these are links for 'Log Analytics Gateway' and 'Learn more about Log Analytics Gateway'.

## Agent install package

The Log Analytics agent for Linux is composed of multiple packages. The release file contains the following packages, which are available by running the shell bundle with the `--extract` parameter:

PACKAGE	VERSION	DESCRIPTION
omsagent	1.14.19	The Log Analytics Agent for Linux
omsconfig	1.1.1	Configuration agent for the Log Analytics agent

PACKAGE	VERSION	DESCRIPTION
omi	1.6.9	Open Management Infrastructure (OMI) -- a lightweight CIM Server. <i>Note that OMI requires root access to run a cron job necessary for the functioning of the service</i>
scx	1.6.9	OMI CIM Providers for operating system performance metrics
apache-cimprov	1.0.1	Apache HTTP Server performance monitoring provider for OMI. Only installed if Apache HTTP Server is detected.
mysql-cimprov	1.0.1	MySQL Server performance monitoring provider for OMI. Only installed if MySQL/MariaDB server is detected.
docker-cimprov	1.0.0	Docker provider for OMI. Only installed if Docker is detected.

## Agent installation details

### IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

Installing the Log Analytics agent for Linux packages also applies the system-wide configuration changes below. Uninstalling the `omsagent` package removes these artifacts.

- A non-privileged user named: `omsagent` is created. The daemon runs under this credential.
- A sudoers *include* file is created in `/etc/sudoers.d/omsagent`. This authorizes `omsagent` to restart the syslog and omsagent daemons. If sudo *include* directives aren't supported in the installed version of sudo, these entries will be written to `/etc/sudoers`.
- The syslog configuration is modified to forward a subset of events to the agent. For more information, see [Configure Syslog data collection](#).

On a monitored Linux computer, the agent is listed as `omsagent`. `omsconfig` is the Log Analytics agent for Linux configuration agent that looks for new portal side configuration every 5 minutes. The new and updated configuration is applied to the agent configuration files located at `/etc/opt/microsoft/omsagent/conf/omsagent.conf`.

## Install the agent

### IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

- [Wrapper script](#)

- [Shell](#)

The following steps configure setup of the agent for Log Analytics in Azure and Azure Government cloud using the wrapper script for Linux computers that can communicate directly or through a proxy server to download the agent hosted on GitHub and install the agent.

If your Linux computer needs to communicate through a proxy server to Log Analytics, this configuration can be specified on the command line by including `-p [protocol://][user:password@]proxyhost[:port]`. The *protocol* property accepts `http` or `https`, and the *proxyhost* property accepts a fully qualified domain name or IP address of the proxy server.

For example: `https://proxy01.contoso.com:30443`

If authentication is required in either case, you need to specify the username and password. For example:

`https://user01:password@proxy01.contoso.com:30443`

1. To configure the Linux computer to connect to a Log Analytics workspace, run the following command providing the workspace ID and primary key. The following command downloads the agent, validates its checksum, and installs it.

```
wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -w <YOUR WORKSPACE ID> -s <YOUR WORKSPACE PRIMARY KEY>
```

The following command includes the `-p` proxy parameter and example syntax when authentication is required by your proxy server:

```
wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -p [protocol://]<proxy user>:<proxy password>@<proxyhost>[:port] -w <YOUR WORKSPACE ID> -s <YOUR WORKSPACE PRIMARY KEY>
```

2. To configure the Linux computer to connect to Log Analytics workspace in Azure Government cloud, run the following command providing the workspace ID and primary key copied earlier. The following command downloads the agent, validates its checksum, and installs it.

```
wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -w <YOUR WORKSPACE ID> -s <YOUR WORKSPACE PRIMARY KEY> -d opinsights.azure.us
```

The following command includes the `-p` proxy parameter and example syntax when authentication is required by your proxy server:

```
wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh -p [protocol://]<proxy user>:<proxy password>@<proxyhost>[:port] -w <YOUR WORKSPACE ID> -s <YOUR WORKSPACE PRIMARY KEY> -d opinsights.azure.us
```

3. Restart the agent by running the following command:

```
sudo /opt/microsoft/omsagent/bin/service_control restart [<workspace id>]
```

## Upgrade from a previous release

Upgrading from a previous version, starting with version 1.0.0-47, is supported in each release. Perform the installation with the `--upgrade` parameter to upgrade all components of the agent to the latest version.

#### NOTE

There will be a warning message during the upgrade "docker provider package installation skipped" since `--skip-docker-provider-install` flag is set. If you are installing over an existing omsagent install and wish to remove the docker provider, you should first purge the existing installation and then install using the `--skip-docker-provider-install` flag.

## Cache information

Data from the Log Analytics agent for Linux is cached on the local machine at `%STATE_DIR_WS%/out_oms_common.buffer*` before it's sent to Azure Monitor. Custom log data is buffered in `%STATE_DIR_WS%/out_oms_blob.buffer*`. The path may be different for some [solutions and data types](#).

The agent attempts to upload every 20 seconds. If it fails, it waits an exponentially increasing length of time until it succeeds: 30 seconds before the second attempt, 60 seconds before the third, 120 seconds, and so on, up to a maximum of 16 minutes between retries until it successfully connects again. The agent retries up to 6 times for a given chunk of data before discarding and moving to the next one. This continues until the agent can successfully upload again. This means that data may be buffered up to approximately 30 minutes before being discarded.

The default cache size is 10 MB but can be modified in the [omsagent.conf file](#).

## Next steps

- Review [Managing and maintaining the Log Analytics agent for Windows and Linux](#) to learn about how to reconfigure, upgrade, or remove the agent from the virtual machine.
- Review [Troubleshooting the Linux agent](#) if you encounter issues while installing or managing the agent.
- Review [Agent Data Sources](#) to learn about data source configuration.

# Connect computers without internet access by using the Log Analytics gateway in Azure Monitor

9/21/2022 • 21 minutes to read • [Edit Online](#)

This article describes how to configure communication with Azure Automation and Azure Monitor by using the Log Analytics gateway when computers that are directly connected or that are monitored by Operations Manager have no internet access.

The Log Analytics gateway is an HTTP forward proxy that supports HTTP tunneling using the HTTP CONNECT command. This gateway sends data to Azure Automation and a Log Analytics workspace in Azure Monitor on behalf of the computers that cannot directly connect to the internet. The gateway is only for log agent related connectivity and does not support Azure Automation features like runbook, DSC, and others.

The Log Analytics gateway supports:

- Reporting up to the same Log Analytics workspaces configured on each agent behind it and that are configured with Azure Automation Hybrid Runbook Workers.
- Windows computers on which either the [Azure Monitor Agent](#) or the legacy Microsoft Monitoring Agent is directly connected to a Log Analytics workspace in Azure Monitor. Both the source and the gateway server must be running the same agent. You can't stream events from a server running Azure Monitor agent through a server running the gateway with the Log Analytics agent.
- Linux computers on which either the [Azure Monitor Agent](#) or the legacy Log Analytics agent for Linux is directly connected to a Log Analytics workspace in Azure Monitor.
- System Center Operations Manager 2012 SP1 with UR7, Operations Manager 2012 R2 with UR3, or a management group in Operations Manager 2016 or later that is integrated with Log Analytics.

Some IT security policies don't allow internet connection for network computers. These unconnected computers could be point of sale (POS) devices or servers supporting IT services, for example. To connect these devices to Azure Automation or a Log Analytics workspace so you can manage and monitor them, configure them to communicate directly with the Log Analytics gateway. The Log Analytics gateway can receive configuration information and forward data on their behalf. If the computers are configured with the Log Analytics agent to directly connect to a Log Analytics workspace, the computers instead communicate with the Log Analytics gateway.

The Log Analytics gateway transfers data from the agents to the service directly. It doesn't analyze any of the data in transit and the gateway does not cache data when it loses connectivity with the service. When the gateway is unable to communicate with service, the agent continues to run and queues the collected data on the disk of the monitored computer. When the connection is restored, the agent sends the cached data collected to Azure Monitor.

When an Operations Manager management group is integrated with Log Analytics, the management servers can be configured to connect to the Log Analytics gateway to receive configuration information and send collected data, depending on the solution you have enabled. Operations Manager agents send some data to the management server. For example, agents might send Operations Manager alerts, configuration assessment data, instance space data, and capacity data. Other high-volume data, such as Internet Information Services (IIS) logs, performance data, and security events, is sent directly to the Log Analytics gateway.

If one or more Operations Manager Gateway servers are deployed to monitor untrusted systems in a perimeter network or an isolated network, those servers can't communicate with a Log Analytics gateway. Operations Manager Gateway servers can report only to a management server. When an Operations Manager management

group is configured to communicate with the Log Analytics gateway, the proxy configuration information is automatically distributed to every agent-managed computer that is configured to collect log data for Azure Monitor, even if the setting is empty.

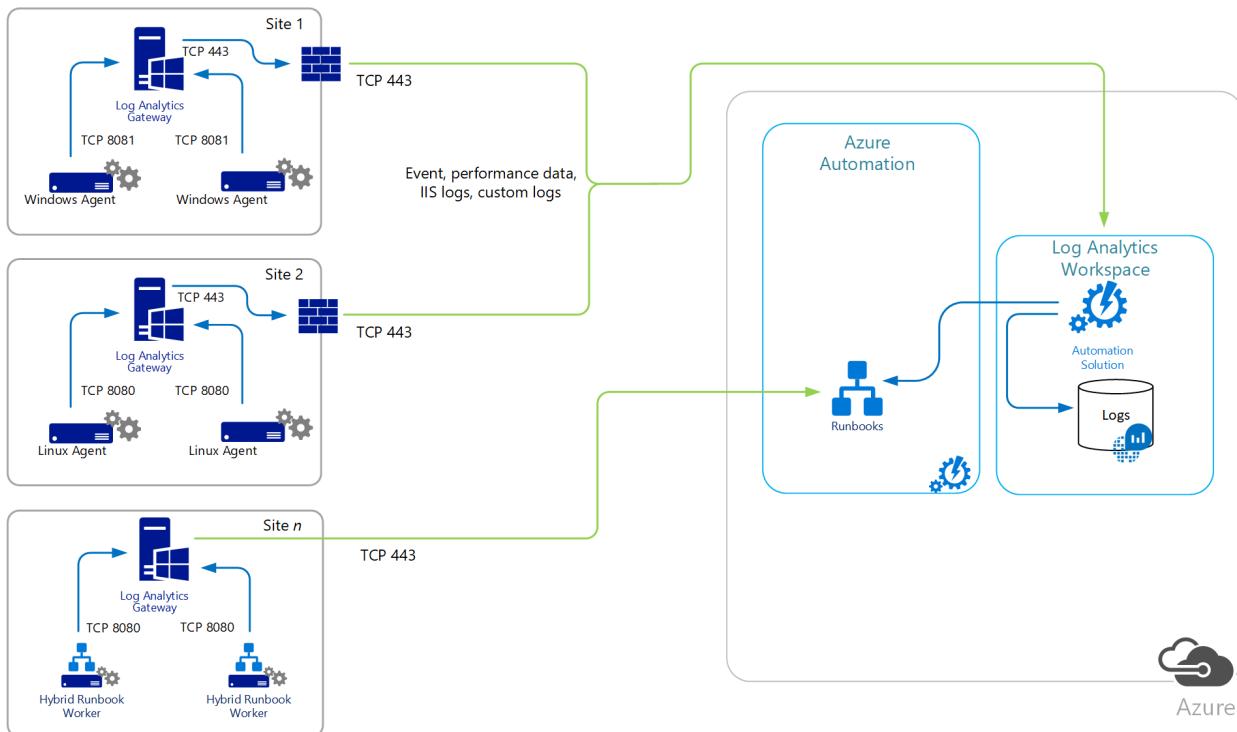
To provide high availability for directly connected or Operations Management groups that communicate with a Log Analytics workspace through the gateway, use network load balancing (NLB) to redirect and distribute traffic across multiple gateway servers. That way, if one gateway server goes down, the traffic is redirected to another available node.

The computer that runs the Log Analytics gateway requires the agent to identify the service endpoints that the gateway needs to communicate with. The agent also needs to direct the gateway to report to the same workspaces that the agents or Operations Manager management group behind the gateway are configured with. This configuration allows the gateway and the agent to communicate with their assigned workspace.

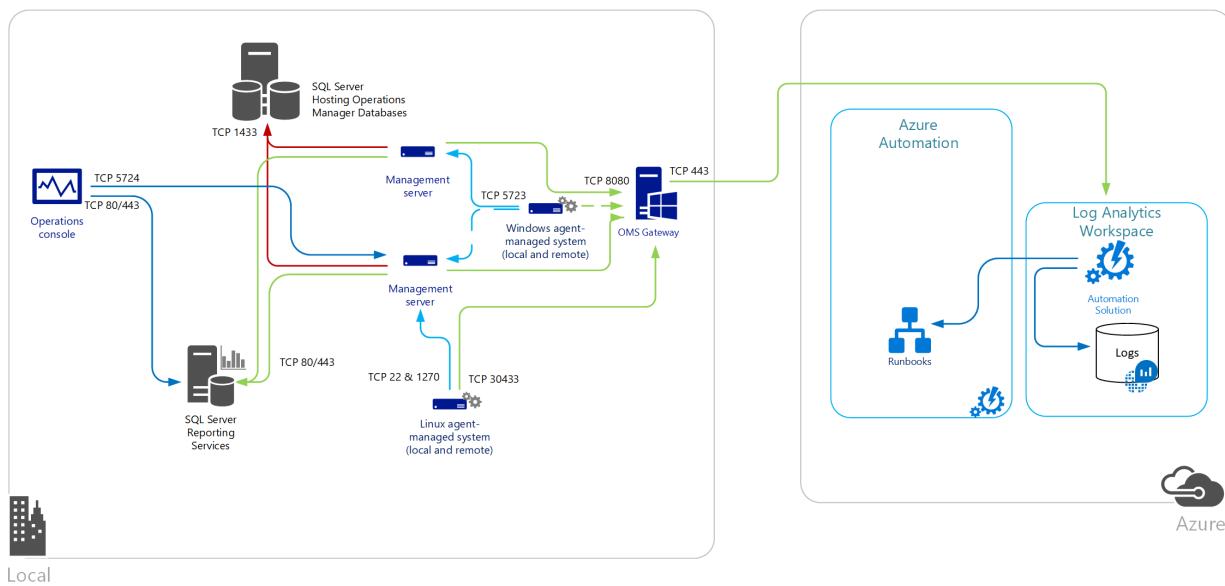
A gateway can be multihomed to up to ten workspaces using the Azure Monitor Agent and [data collection rules](#). Using the legacy Microsoft Monitor Agent, you can only multihome up to four workspaces as that is the total number of workspaces the legacy Windows agent supports.

Each agent must have network connectivity to the gateway so that agents can automatically transfer data to and from the gateway. Avoid installing the gateway on a domain controller. Linux computers that are behind a gateway server cannot use the [wrapper script installation](#) method to install the Log Analytics agent for Linux. The agent must be downloaded manually, copied to the computer, and installed manually because the gateway only supports communicating with the Azure services mentioned earlier.

The following diagram shows data flowing from direct agents, through the gateway, to Azure Automation and Log Analytics. The agent proxy configuration must match the port that the Log Analytics gateway is configured with.



The following diagram shows data flow from an Operations Manager management group to Log Analytics.



## Set up your system

Computers designated to run the Log Analytics gateway must have the following configuration:

- Windows 10, Windows 8.1, or Windows 7
- Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows Server 2008 R2, or Windows Server 2008
- Microsoft .NET Framework 4.5
- At least a 4-core processor and 8 GB of memory
- An [Azure Monitor agent](#) installed with [data collection rule\(s\)](#) configured, or the [Log Analytics agent for Windows](#) configured to report to the same workspace as the agents that communicate through the gateway

### Language availability

The Log Analytics gateway is available in these languages:

- Chinese (Simplified)
- Chinese (Traditional)
- Czech
- Dutch
- English
- French
- German
- Hungarian
- Italian
- Japanese
- Korean
- Polish
- Portuguese (Brazil)
- Portuguese (Portugal)
- Russian
- Spanish (International)

### Supported encryption protocols

The Log Analytics gateway supports only Transport Layer Security (TLS) 1.0, 1.1, and 1.2. It doesn't support Secure Sockets Layer (SSL). To ensure the security of data in transit to Log Analytics, configure the gateway to

use at least TLS 1.2. Older versions of TLS or SSL are vulnerable. Although they currently allow backward compatibility, avoid using them.

For additional information, review [Sending data securely using TLS 1.2](#).

**NOTE**

The gateway is a forwarding proxy that doesn't store any data. Once the agent establishes connection with Azure Monitor, it follows the same encryption flow with or without the gateway. The data is encrypted between the client and the endpoint. Since the gateway is just a tunnel, it doesn't have the ability to inspect what is being sent.

## Supported number of agent connections

The following table shows approximately how many agents can communicate with a gateway server. Support is based on agents that upload about 200 KB of data every 6 seconds. For each agent tested, data volume is about 2.7 GB per day.

GATEWAY	AGENTS SUPPORTED (APPROXIMATE)
CPU: Intel Xeon Processor E5-2660 v3 @ 2.6 GHz 2 Cores Memory: 4 GB Network bandwidth: 1 Gbps	600
CPU: Intel Xeon Processor E5-2660 v3 @ 2.6 GHz 4 Cores Memory: 8 GB Network bandwidth: 1 Gbps	1000

## Download the Log Analytics gateway

Get the latest version of the Log Analytics gateway Setup file from either Microsoft Download Center ([Download Link](#)) or the Azure portal.

To get the Log Analytics gateway from the Azure portal, follow these steps:

1. Browse the list of services, and then select **Log Analytics**.
2. Select a workspace.
3. In your workspace blade, under **General**, select **Quick Start**.
4. Under **Choose a data source to connect to the workspace**, select **Computers**.
5. In the **Direct Agent** blade, select **Download Log Analytics gateway**.

The screenshot shows the 'bandersfree - Quick Start' Log Analytics workspace. On the left, there's a sidebar with 'General' settings like Overview, Saved searches, Log Search, Solutions, Pricing tier, Log Analytics usage, and Properties. The 'Quick Start' button is highlighted with a red box. The main area displays options to 'Choose a data source to connect to the workspace' (Azure virtual machines (VMs), Storage account logs, Computers, System Center Operations Manager) and 'View collected data' (OMS Portal). Below that, there's a 'Learn more' section with 'Help documentation' and 'Give feedback' links.

**Direct Agent**

bandersfree

Download an agent for your operating system, then install and configure it using the keys for your workspace ID.

Windows Computers

[Download Windows Agent \(64 bit\)](#)  
[Download Windows Agent \(32 bit\)](#)

Linux Computers

Agent for Linux  
DOWNLOAD AND ONBOARD AGENT FOR LINUX  
`wget https://raw.githubusercontent.com/`

Workspace ID and Keys

WORKSPACE ID  
`d61b4bcf-0000-0000-0000-000000000000`

PRIMARY KEY  
`SgYGiixRFwT10nHv1Ow5oCSCWm80T5j`

SECONDARY KEY  
`6qra+9tGCKhnhMzL7y+TM99d9ZM6Aa`

OMS Gateway

If you have machines with no internet connectivity to OMS, download the OMS Gateway to act as a proxy. [Learn more](#).  
[Download OMS Gateway](#)

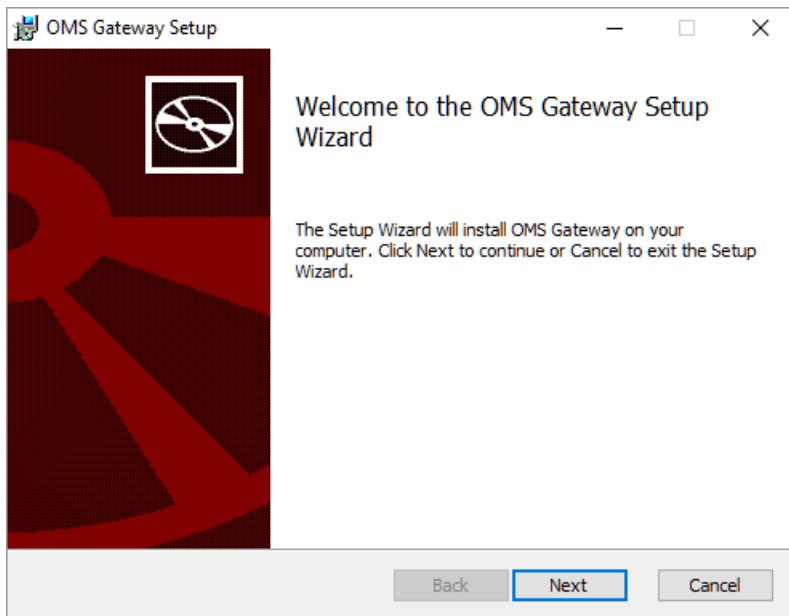
or

1. In your workspace blade, under Settings, select Advanced settings.
2. Go to Connected Sources > Windows Servers and select Download Log Analytics gateway.

## Install Log Analytics gateway using setup wizard

To install a gateway using the setup wizard, follow these steps.

1. From the destination folder, double-click Log Analytics gateway.msi.
2. On the Welcome page, select Next.

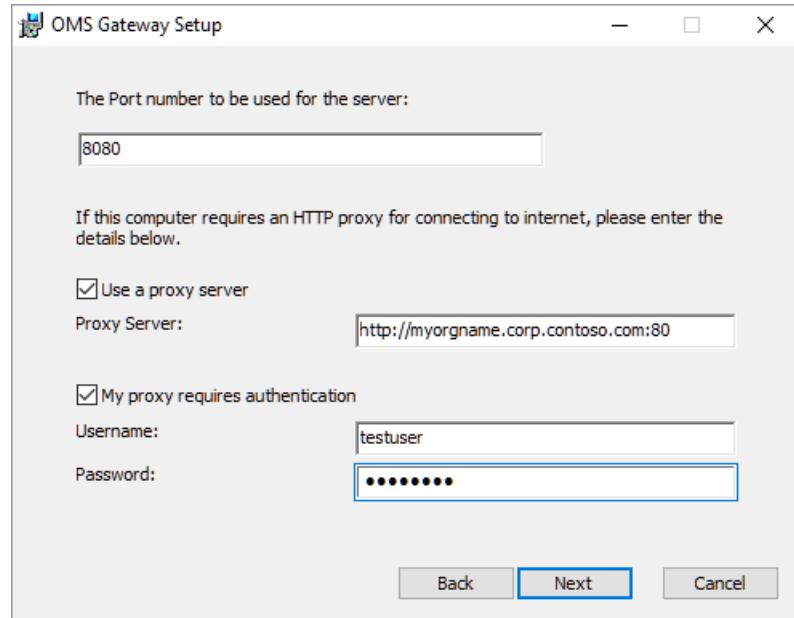


3. On the License Agreement page, select I accept the terms in the License Agreement to agree to the Microsoft Software License Terms, and then select Next.
4. On the Port and proxy address page:
  - a. Enter the TCP port number to be used for the gateway. Setup uses this port number to configure an

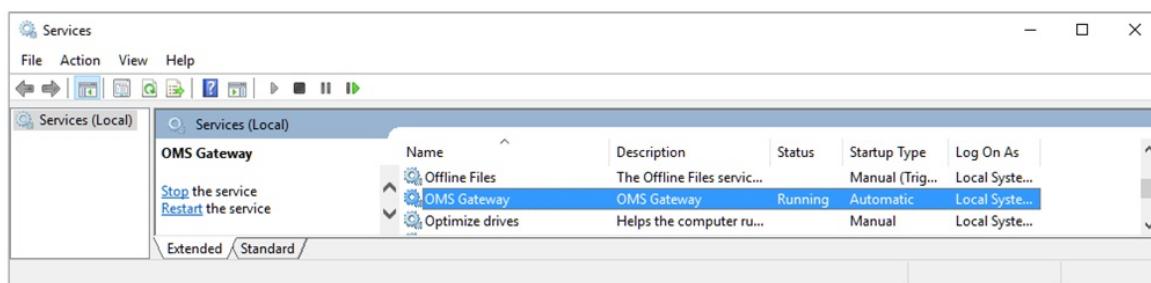
inbound rule on Windows Firewall. The default value is 8080. The valid range of the port number is 1 through 65535. If the input does not fall into this range, an error message appears.

b. If the server where the gateway is installed needs to communicate through a proxy, enter the proxy address where the gateway needs to connect. For example, enter `http://myorgname.corp.contoso.com:80`. If you leave this field blank, the gateway will try to connect to the internet directly. If your proxy server requires authentication, enter a username and password.

c. Select **Next**.



5. If you do not have Microsoft Update enabled, the Microsoft Update page appears, and you can choose to enable it. Make a selection and then select **Next**. Otherwise, continue to the next step.
6. On the **Destination Folder** page, either leave the default folder C:\Program Files\OMS Gateway or enter the location where you want to install the gateway. Then select **Next**.
7. On the **Ready to install** page, select **Install**. If User Account Control requests permission to install, select **Yes**.
8. After Setup finishes, select **Finish**. To verify that the service is running, open the services.msc snap-in and verify that **OMS Gateway** appears in the list of services and that its status is **Running**.



## Install the Log Analytics gateway using the command line

The downloaded file for the gateway is a Windows Installer package that supports silent installation from the command line or other automated method. If you are not familiar with the standard command-line options for Windows Installer, see [Command-line options](#).

The following table highlights the parameters supported by setup.

PARAMETERS	NOTES
PORNUMBER	TCP port number for gateway to listen on
PROXY	IP address of proxy server
INSTALLDIR	Fully qualified path to specify install directory of gateway software files
USERNAME	User ID to authenticate with proxy server
PASSWORD	Password of the user ID to authenticate with proxy
LicenseAccepted	Specify a value of 1 to verify you accept license agreement
HASAUTH	Specify a value of 1 when USERNAME/PASSWORD parameters are specified
HASPROXY	Specify a value of 1 when specifying IP address for PROXY parameter

To silently install the gateway and configure it with a specific proxy address, port number, type the following:

```
Msieexec.exe /I "oms_gateway.msi" /qn PORTNUMBER=8080 PROXY="10.80.2.200" HASPROXY=1 LicenseAccepted=1
```

Using the /qn command-line option hides setup, /qb shows setup during silent install.

If you need to provide credentials to authenticate with the proxy, type the following:

```
Msieexec.exe /I "oms_gateway.msi" /qn PORTNUMBER=8080 PROXY="10.80.2.200" HASPROXY=1 HASAUTH=1 USERNAME="  
<username>" PASSWORD="  
<password>" LicenseAccepted=1
```

After installation, you can confirm the settings are accepted (excluding the username and password) using the following PowerShell cmdlets:

- **Get-OMSGatewayConfig** – Returns the TCP Port the gateway is configured to listen on.
- **Get-OMSGatewayRelayProxy** – Returns the IP address of the proxy server you configured it to communicate with.

## Configure network load balancing

You can configure the gateway for high availability using network load balancing (NLB) using either Microsoft [Network Load Balancing \(NLB\)](#), [Azure Load Balancer](#), or hardware-based load balancers. The load balancer manages traffic by redirecting the requested connections from the Log Analytics agents or Operations Manager management servers across its nodes. If one Gateway server goes down, the traffic gets redirected to other nodes.

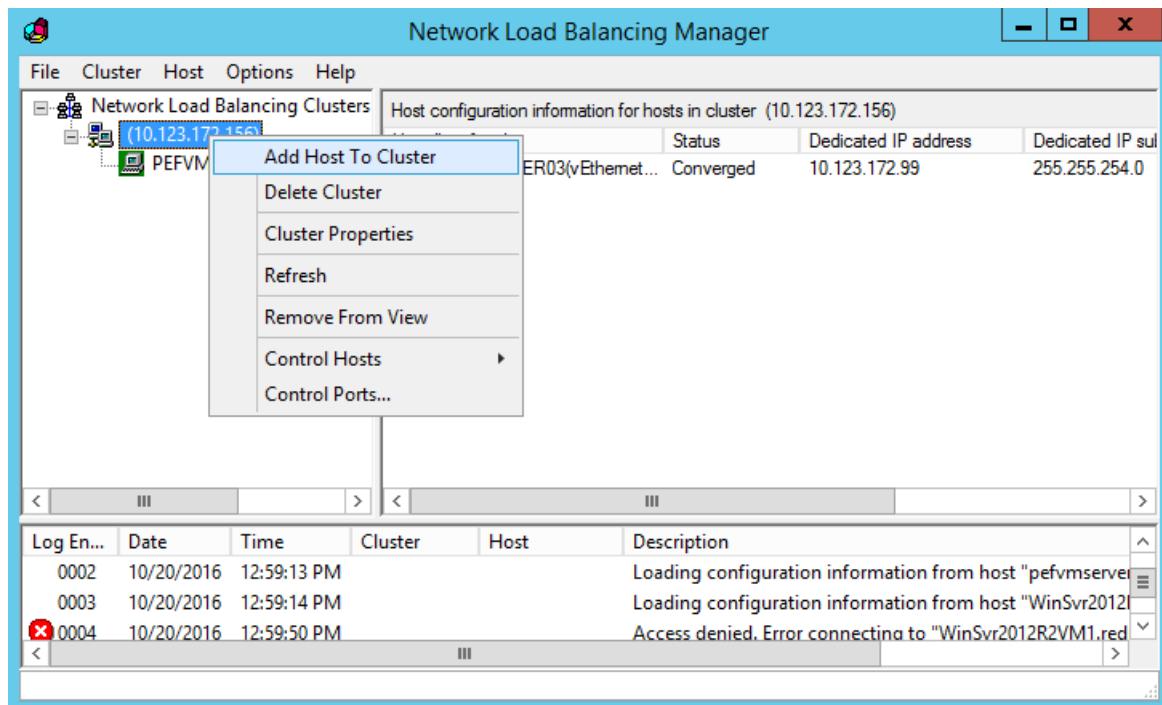
### Microsoft Network Load Balancing

To learn how to design and deploy a Windows Server 2016 network load balancing cluster, see [Network load balancing](#). The following steps describe how to configure a Microsoft network load balancing cluster.

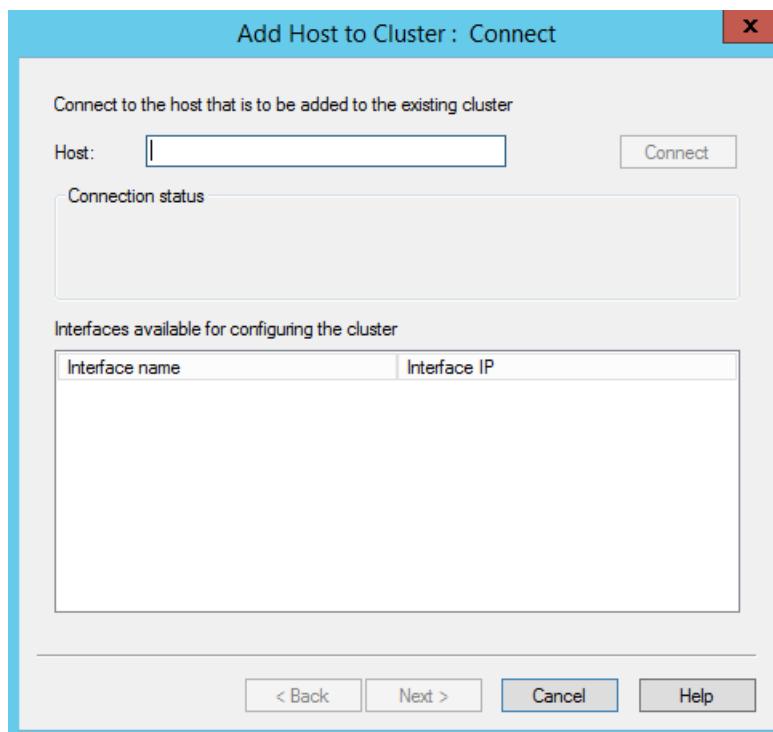
1. Sign onto the Windows server that is a member of the NLB cluster with an administrative account.
2. Open Network Load Balancing Manager in Server Manager, click **Tools**, and then click **Network Load**

## Balancing Manager.

- To connect a Log Analytics gateway server with the Microsoft Monitoring Agent installed, right-click the cluster's IP address, and then click **Add Host to Cluster**.



- Enter the IP address of the gateway server that you want to connect.



## Azure Load Balancer

To learn how to design and deploy an Azure Load Balancer, see [What is Azure Load Balancer?](#). To deploy a basic load balancer, follow the steps outlined in this [quickstart](#) excluding the steps outlined in the section **Create back-end servers**.

#### NOTE

Configuring the Azure Load Balancer using the **Basic SKU**, requires that Azure virtual machines belong to an Availability Set. To learn more about availability sets, see [Manage the availability of Windows virtual machines in Azure](#). To add existing virtual machines to an availability set, refer to [Set Azure Resource Manager VM Availability Set](#).

After the load balancer is created, a backend pool needs to be created, which distributes traffic to one or more gateway servers. Follow the steps described in the quickstart article section [Create resources for the load balancer](#).

#### NOTE

When configuring the health probe it should be configured to use the TCP port of the gateway server. The health probe dynamically adds or removes the gateway servers from the load balancer rotation based on their response to health checks.

## Configure the Azure Monitor agent to communicate using Log Analytics gateway

To configure the Azure Monitor agent (installed on the gateway server) to use the gateway to upload data for Windows or Linux:

1. Follow the instructions to [configure proxy settings on the agent](#) and provide the IP address and port number corresponding to the gateway server. If you have deployed multiple gateway servers behind a load balancer, the agent proxy configuration is the virtual IP address of the load balancer instead.
2. Add the **configuration endpoint URL** to fetch data collection rules to the allow list for the gateway  

```
Add-OMSGatewayAllowedHost -Host global.handler.control.monitor.azure.com
```

```
Add-OMSGatewayAllowedHost -Host <gateway-server-region-name>.handler.control.monitor.azure.com
```

(If using private links on the agent, you must also add the [dce endpoints](#))
3. Add the **data ingestion endpoint URL** to the allow list for the gateway  

```
Add-OMSGatewayAllowedHost -Host <log-analytics-workspace-id>.ods.opinsights.azure.com
```
4. Restart the **OMS Gateway** service to apply the changes  

```
Stop-Service -Name <gateway-name>
```

```
Start-Service -Name <gateway-name>
```

## Configure the Log Analytics agent and Operations Manager management group

In this section, you'll see how to configure directly connected legacy Log Analytics agents, an Operations Manager management group, or Azure Automation Hybrid Runbook Workers with the Log Analytics gateway to communicate with Azure Automation or Log Analytics.

### Configure a standalone Log Analytics agent

When configuring the legacy Log Analytics agent, replace the proxy server value with the IP address of the Log Analytics gateway server and its port number. If you have deployed multiple gateway servers behind a load balancer, the Log Analytics agent proxy configuration is the virtual IP address of the load balancer.

**NOTE**

To install the Log Analytics agent on the gateway and Windows computers that directly connect to Log Analytics, see [Connect Windows computers to the Log Analytics service in Azure](#). To connect Linux computers, see [Connect Linux computers to Azure Monitor](#).

After you install the agent on the gateway server, configure it to report to the workspace or workspace agents that communicate with the gateway. If the Log Analytics Windows agent is not installed on the gateway, event 300 is written to the OMS Gateway event log, indicating that the agent needs to be installed. If the agent is installed but not configured to report to the same workspace as the agents that communicate through it, event 105 is written to the same log, indicating that the agent on the gateway needs to be configured to report to the same workspace as the agents that communicate with the gateway.

After you complete configuration, restart the **OMS Gateway** service to apply the changes. Otherwise, the gateway will reject agents that attempt to communicate with Log Analytics and will report event 105 in the OMS Gateway event log. This will also happen when you add or remove a workspace from the agent configuration on the gateway server.

For information related to the Automation Hybrid Runbook Worker, see [Automate resources in your datacenter or cloud by using Hybrid Runbook Worker](#).

**Configure Operations Manager, where all agents use the same proxy server**

The Operations Manager proxy configuration is automatically applied to all agents that report to Operations Manager, even if the setting is empty.

To use OMS Gateway to support Operations Manager, you must have:

- Microsoft Monitoring Agent (version 8.0.10900.0 or later) installed on the OMS Gateway server and configured with the same Log Analytics workspaces that your management group is configured to report to.
- Internet connectivity. Alternatively, OMS Gateway must be connected to a proxy server that is connected to the internet.

**NOTE**

If you specify no value for the gateway, blank values are pushed to all agents.

If your Operations Manager management group is registering with a Log Analytics workspace for the first time, you won't see the option to specify the proxy configuration for the management group in the Operations console. This option is available only if the management group has been registered with the service.

To configure integration, update the system proxy configuration by using Netsh on the system where you're running the Operations console and on all management servers in the management group. Follow these steps:

1. Open an elevated command prompt:

- a. Select **Start** and enter **cmd**.
- b. Right-click **Command Prompt** and select **Run as administrator**.

2. Enter the following command:

```
netsh winhttp set proxy <proxy>:<port>
```

After completing the integration with Log Analytics, remove the change by running `netsh winhttp reset proxy`. Then, in the Operations console, use the **Configure proxy server** option to specify the Log Analytics gateway server.

1. On the Operations Manager console, under **Operations Management Suite**, select **Connection**, and then select **Configure Proxy Server**.

The screenshot shows the left navigation pane of the Operations Manager console. Under the 'Operations Management Suite' section, the 'Connection' item is highlighted. In the main content area, the title 'Operations Management Suite Overview' is displayed above an 'Introduction' section. Below the introduction, there is a 'Optional Configuration:' section containing a link to 'Configure Proxy Server'. The 'Configure Proxy Server' link is underlined and has a small blue arrow icon to its left.

2. Select **Use a proxy server to access the Operations Management Suite** and then enter the IP address of the Log Analytics gateway server or virtual IP address of the load balancer. Be careful to start with the prefix `http://`.

The screenshot shows the 'Operations Management Suite Settings Wizard: Proxy Server' window. The title bar says 'Wizard' and 'Operations Management Suite Settings Wizard: Proxy Server'. The left sidebar has a 'Proxy Server' tab selected. The main area contains the following text: 'Web Proxy Server: Define how the management server communicates with the service.' Below this is a checkbox labeled 'Use a proxy server to access the Operations Management Suite' which is checked. To its right is a 'Address:' label and a text input field containing 'http://10.123.173.30:8080'. At the bottom of the window are buttons for '< Previous', 'Next >', 'Finish' (which is highlighted in blue), and 'Cancel'.

3. Select **Finish**. Your Operations Manager management group is now configured to communicate through the gateway server to the Log Analytics service.

#### **Configure Operations Manager, where specific agents use a proxy server**

For large or complex environments, you might want only specific servers (or groups) to use the Log Analytics

gateway server. For these servers, you can't update the Operations Manager agent directly because this value is overwritten by the global value for the management group. Instead, override the rule used to push these values.

#### NOTE

Use this configuration technique if you want to allow for multiple Log Analytics gateway servers in your environment. For example, you can require specific Log Analytics gateway servers to be specified on a regional basis.

To configure specific servers or groups to use the Log Analytics gateway server:

1. Open the Operations Manager console and select the **Authoring** workspace.
2. In the Authoring workspace, select **Rules**.
3. On the Operations Manager toolbar, select the **Scope** button. If this button is not available, make sure you have selected an object, not a folder, in the **Monitoring** pane. The **Scope Management Pack Objects** dialog box displays a list of common targeted classes, groups, or objects.
4. In the **Look for** field, enter **Health Service** and select it from the list. Select **OK**.
5. Search for **Advisor Proxy Setting Rule**.
6. On the Operations Manager toolbar, select **Overrides** and then point to **Override the Rule\For a specific object of class: Health Service** and select an object from the list. Or create a custom group that contains the health service object of the servers you want to apply this override to. Then apply the override to your custom group.
7. In the **Override Properties** dialog box, add a check mark in the **Override** column next to the **WebProxyAddress** parameter. In the **Override Value** field, enter the URL of the Log Analytics gateway server. Be careful to start with the prefix `http://`.

#### NOTE

You don't need to enable the rule. It's already managed automatically with an override in the Microsoft System Center Advisor Secure Reference Override management pack that targets the Microsoft System Center Advisor Monitoring Server Group.

8. Select a management pack from the **Select destination management pack** list, or create a new unsealed management pack by selecting **New**.
9. When you finish, select **OK**.

### Configure for Automation Hybrid Runbook Workers

If you have Automation Hybrid Runbook Workers in your environment, follow these steps to configure the gateway to support the workers.

Refer to the [Configure your network](#) section of the Automation documentation to find the URL for each region.

If your computer is registered as a Hybrid Runbook Worker automatically, for example if the Update Management solution is enabled for one or more VMs, follow these steps:

1. Add the Job Runtime Data service URLs to the Allowed Host list on the Log Analytics gateway. For example:  
`Add-OMSGatewayAllowedHost we-jobruntimedata-prod-su1.azure-automation.net`
2. Restart the Log Analytics gateway service by using the following PowerShell cmdlet:  
`Restart-Service OMSGatewayService`

If your computer is joined to Azure Automation by using the Hybrid Runbook Worker registration cmdlet, follow

these steps:

1. Add the agent service registration URL to the Allowed Host list on the Log Analytics gateway. For example:

```
Add-OMSGatewayAllowedHost ncus-agentservice-prod-1.azure-automation.net
```

2. Add the Job Runtime Data service URLs to the Allowed Host list on the Log Analytics gateway. For example:

```
Add-OMSGatewayAllowedHost we-jobruntimedata-prod-su1.azure-automation.net
```

3. Restart the Log Analytics gateway service.

```
Restart-Service OMSGatewayService
```

## Useful PowerShell cmdlets

You can use cmdlets to complete the tasks to update the Log Analytics gateway's configuration settings. Before you use cmdlets, be sure to:

1. Install the Log Analytics gateway (Microsoft Windows Installer).
2. Open a PowerShell console window.
3. Import the module by typing this command:

```
Import-Module OMSGateway
```

4. If no error occurred in the previous step, the module was successfully imported and the cmdlets can be used.

Enter

```
Get-Module OMSGateway
```

5. After you use the cmdlets to make changes, restart the OMS Gateway service.

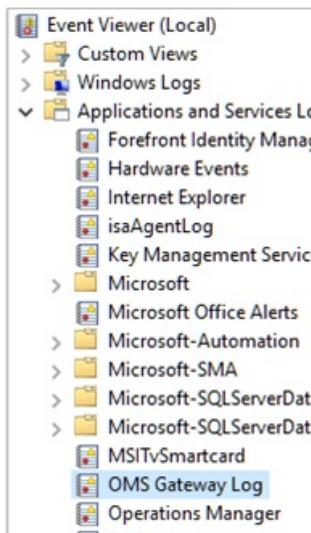
An error in step 3 means that the module wasn't imported. The error might occur when PowerShell can't find the module. You can find the module in the OMS Gateway installation path: *C:\Program Files\Microsoft OMS Gateway\PowerShell\OmsGateway*.

CMDLET	PARAMETERS	DESCRIPTION	EXAMPLE
<code>Get-OMSGatewayConfig</code>	Key	Gets the configuration of the service	<code>Get-OMSGatewayConfig</code>
<code>Set-OMSGatewayConfig</code>	Key (required) Value	Changes the configuration of the service	<code>Set-OMSGatewayConfig -Name ListenPort -Value 8080</code>
<code>Get-OMSGatewayRelayProxy</code>		Gets the address of relay (upstream) proxy	<code>Get-OMSGatewayRelayProxy</code>
<code>Set-OMSGatewayRelayProxy</code>	Address Username Password (secure string)	Sets the address (and credential) of relay (upstream) proxy	<ol style="list-style-type: none"><li>1. Set a relay proxy and credential: <pre>Set-OMSGatewayRelayProxy -Address http://www.myproxy.com:8080 -Username user1 - Password 123</pre></li><li>2. Set a relay proxy that doesn't need authentication: <pre>Set-OMSGatewayRelayProxy -Address http://www.myproxy.com:8080</pre></li><li>3. Clear the relay proxy setting: <pre>Set-OMSGatewayRelayProxy -Address ""</pre></li></ol>

CMDLET	PARAMETERS	DESCRIPTION	EXAMPLE
Get-OMSGatewayAllowedHost		Gets the currently allowed host (only the locally configured allowed host, not automatically downloaded allowed hosts)	Get-OMSGatewayAllowedHost
Add-OMSGatewayAllowedHost	Host (required)	Adds the host to the allowed list	Add-OMSGatewayAllowedHost -Host www.test.com
Remove-OMSGatewayAllowedHost	Host (required)	Removes the host from the allowed list	Remove-OMSGatewayAllowedHost -Host www.test.com
Add-OMSGatewayAllowedClientCertificate	Subject (required)	Adds the client certificate subject to the allowed list	Add-OMSGatewayAllowed ClientCertificate -Subject mycert
Remove-OMSGatewayAllowedClientCertificate	Subject (required)	Removes the client certificate subject from the allowed list	Remove-OMSGatewayAllowed ClientCertificate -Subject mycert
Get-OMSGatewayAllowedClientCertificate		Gets the currently allowed client certificate subjects (only the locally configured allowed subjects, not automatically downloaded allowed subjects)	Get-OMSGatewayAllowed ClientCertificate

## Troubleshooting

To collect events logged by the gateway, you should have the Log Analytics agent installed.



### Log Analytics gateway event IDs and descriptions

The following table shows the event IDs and descriptions for Log Analytics gateway log events.

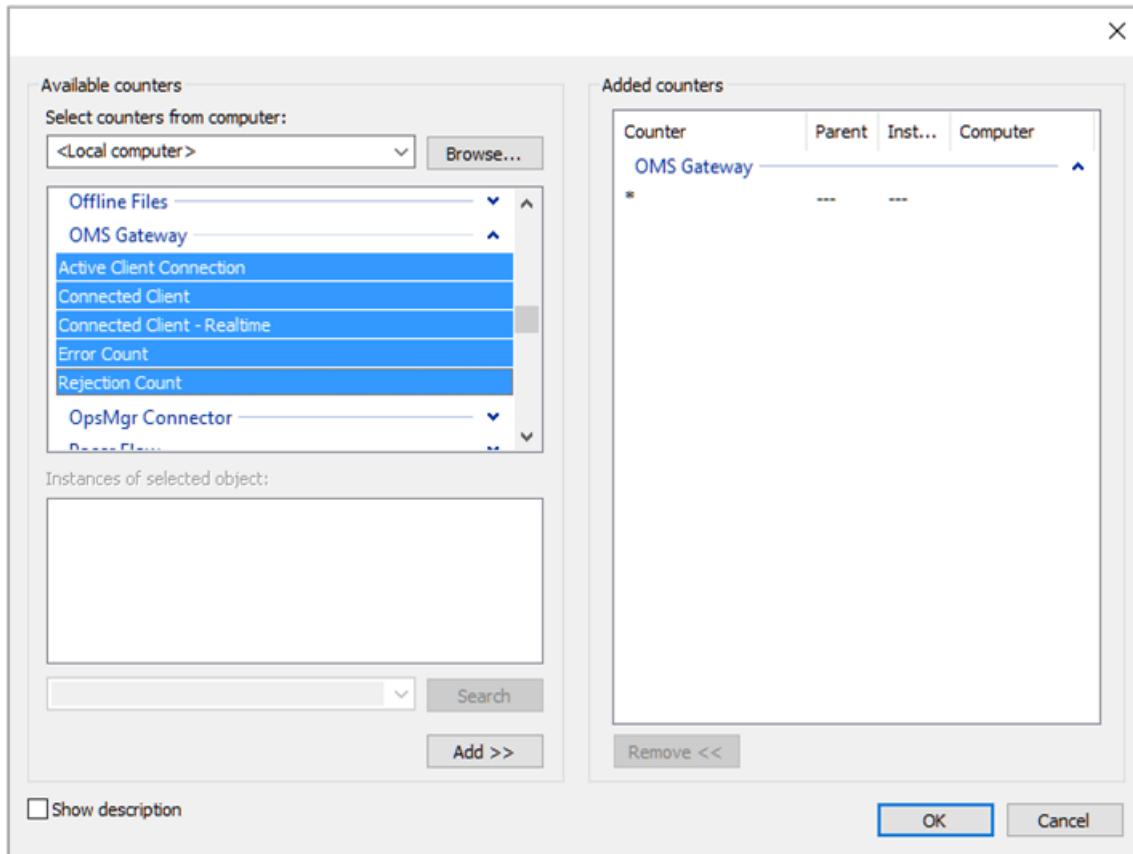
ID	DESCRIPTION
400	Any application error that has no specific ID.
401	Wrong configuration. For example, listenPort = "text" instead of an integer.
402	Exception in parsing TLS handshake messages.
403	Networking error. For example, cannot connect to target server.
100	General information.
101	Service has started.
102	Service has stopped.
103	An HTTP CONNECT command was received from client.
104	Not an HTTP CONNECT command.
105	Destination server is not in allowed list, or destination port is not secure (443).  Ensure that the MMA agent on your OMS Gateway server and the agents that communicate with OMS Gateway are connected to the same Log Analytics workspace.
105	ERROR TcpConnection – Invalid Client certificate: CN=Gateway.  Ensure that you're using OMS Gateway version 1.0.395.0 or greater. Also ensure that the MMA agent on your OMS Gateway server and the agents communicating with OMS Gateway are connected to the same Log Analytics workspace.
106	Unsupported TLS/SSL protocol version.  The Log Analytics gateway supports only TLS 1.0, TLS 1.1, and 1.2. It does not support SSL.
107	The TLS session has been verified.

### Performance counters to collect

The following table shows the performance counters available for the Log Analytics gateway. Use Performance Monitor to add the counters.

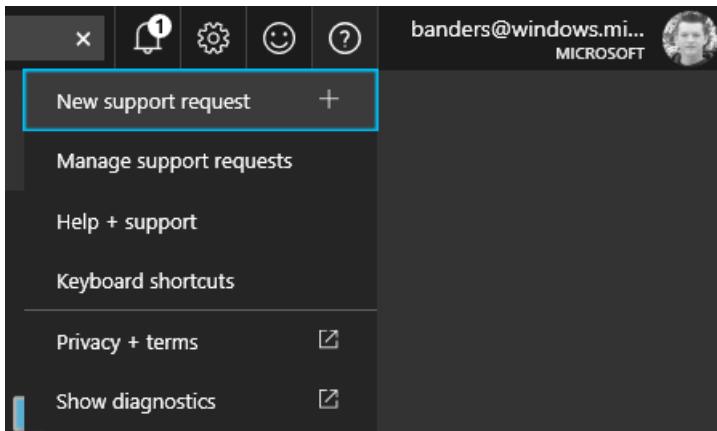
NAME	DESCRIPTION
Log Analytics Gateway/Active Client Connection	Number of active client network (TCP) connections
Log Analytics Gateway/Error Count	Number of errors

NAME	DESCRIPTION
Log Analytics Gateway/Connected Client	Number of connected clients
Log Analytics Gateway/Rejection Count	Number of rejections due to any TLS validation error



## Assistance

When you're signed in to the Azure portal, you can get help with the Log Analytics gateway or any other Azure service or feature. To get help, select the question mark icon in the upper-right corner of the portal and select **New support request**. Then complete the new support request form.



## Next steps

Add [data sources](#) to collect data from connected sources, and store the data in your Log Analytics workspace.

# Manage and maintain the Log Analytics agent for Windows and Linux

9/21/2022 • 11 minutes to read • [Edit Online](#)

After initial deployment of the Log Analytics Windows or Linux agent in Azure Monitor, you may need to reconfigure the agent, upgrade it, or remove it from the computer if it has reached the retirement stage in its lifecycle. You can easily manage these routine maintenance tasks manually or through automation, which reduces both operational error and expenses.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

## Upgrade the agent

Upgrade to the latest release of the Log Analytics agent for Windows and Linux manually or automatically based on your deployment scenario and the environment the VM is running in:

ENVIRONMENT	INSTALLATION METHOD	UPGRADE METHOD
Azure VM	Log Analytics agent VM extension for Windows/Linux	Agent is automatically upgraded <a href="#">after the VM model changes</a> , unless you configured your Azure Resource Manager template to opt out by setting the property <code>autoUpgradeMinorVersion</code> to <code>false</code> . Once deployed, however, the extension will not upgrade minor versions unless redeployed, even with this property set to true. Only Linux agent supports automatic update post deployment with <code>enableAutomaticUpgrade</code> property(See <a href="#">Enable Auto-Update for the Linux Agent</a> ). Major version upgrade is always manual(See <a href="#">VirtualMachineExtensionInner.AutoUpgradeMinorVersion Property</a> ).
Custom Azure VM images	Manual install of Log Analytics agent for Windows/Linux	Updating VMs to the newest version of the agent needs to be performed from the command line running the Windows installer package or Linux self-extracting and installable shell script bundle.
Non-Azure VMs	Manual install of Log Analytics agent for Windows/Linux	Updating VMs to the newest version of the agent needs to be performed from the command line running the Windows installer package or Linux self-extracting and installable shell script bundle.

## Upgrade Windows agent

To update the agent on a Windows VM to the latest version not installed using the Log Analytics VM extension, you either run from the Command Prompt, script or other automation solution, or by using the MMASetup-<platform>.msi Setup Wizard.

You can download the latest version of the Windows agent from your Log Analytics workspace, by performing the following steps.

1. Sign in to the [Azure portal](#).
2. In the Azure portal, click **All services**. In the list of resources, type **Log Analytics**. As you begin typing, the list filters based on your input. Select **Log Analytics workspaces**.
3. In your list of Log Analytics workspaces, select the workspace.
4. In your Log Analytics workspace, select **Agents Management** tile, and then **Windows Servers**.
5. From the **Windows Servers** page, select the appropriate **Download Windows Agent** version to download depending on the processor architecture of the Windows operating system.

### NOTE

During the upgrade of the Log Analytics agent for Windows, it does not support configuring or reconfiguring a workspace to report to. To configure the agent, you need to follow one of the supported methods listed under [Add or remove a workspace](#).

### To upgrade using the Setup Wizard

1. Sign on to the computer with an account that has administrative rights.
2. Execute **MMASetup-<platform>.exe** to start the Setup Wizard.
3. On the first page of the Setup Wizard, click **Next**.
4. In the **Microsoft Monitoring Agent Setup** dialog box, click **I agree** to accept the license agreement.
5. In the **Microsoft Monitoring Agent Setup** dialog box, click **Upgrade**. The status page displays the progress of the upgrade.
6. When the **Microsoft Monitoring Agent configuration completed successfully** page appears, click **Finish**.

### To upgrade from the command line

1. Sign on to the computer with an account that has administrative rights.
2. To extract the agent installation files, from an elevated command prompt run `MMASetup-<platform>.exe /c` and it will prompt you for the path to extract files to. Alternatively, you can specify the path by passing the arguments `MMASetup-<platform>.exe /c /t:<Full Path>`.
3. Run the following command, where D:\ is the location for the upgrade log file.

```
setup.exe /qn /l*v D:\logs\AgentUpgrade.log AcceptEndUserLicenseAgreement=1
```

## Upgrade Linux agent

Upgrade from prior versions (>1.0.0-47) is supported. Performing the installation with the `--upgrade` command will upgrade all components of the agent to the latest version.

Run the following command to upgrade the agent.

```
sudo sh ./omsagent-*.universal.x64.sh --upgrade
```

## Enable Auto-Update for the Linux Agent

We recommend enabling [Automatic Extension Upgrade](#) using these commands to update the agent automatically:

- [Powershell](#)
- [Azure CLI](#)

```
Set-AzVMExtension \
    -ResourceGroupName myResourceGroup \
    -VMName myVM \
    -ExtensionName OmsAgentForLinux \
    -ExtensionType OmsAgentForLinux \
    -Publisher Microsoft.EnterpriseCloud.Monitoring \
    -TypeHandlerVersion latestVersion \
    -ProtectedSettingString '{"workspaceKey":"myWorkspaceKey"}' \
    -SettingString '{"workspaceId":"myWorkspaceId","skipDockerProviderInstall": true}' \
    -EnableAutomaticUpgrade $true
```

## Add or remove a workspace

### Windows agent

The steps in this section are necessary when you want to not only reconfigure the Windows agent to report to a different workspace or to remove a workspace from its configuration, but also when you want to configure the agent to report to more than one workspace (commonly referred to as multi-homing). Configuring the Windows agent to report to multiple workspaces can only be performed after initial setup of the agent and using the methods described below.

#### Update settings from Control Panel

1. Sign on to the computer with an account that has administrative rights.
2. Open **Control Panel**.
3. Select **Microsoft Monitoring Agent** and then click the **Azure Log Analytics** tab.
4. If removing a workspace, select it and then click **Remove**. Repeat this step for any other workspace you want the agent to stop reporting to.
5. If adding a workspace, click **Add** and on the **Add a Log Analytics Workspace** dialog box, paste the Workspace ID and Workspace Key (Primary Key). If the computer should report to a Log Analytics workspace in Azure Government cloud, select Azure US Government from the Azure Cloud drop-down list.
6. Click **OK** to save your changes.

#### Remove a workspace using PowerShell

```
$workspaceId = "<Your workspace Id>"  
$mma = New-Object -ComObject 'AgentConfigManager.MgmtSvcCfg'  
$mma.RemoveCloudWorkspace($workspaceId)  
$mma.ReloadConfiguration()
```

#### Add a workspace in Azure commercial using PowerShell

```
$workspaceId = "<Your workspace Id>"  
$workspaceKey = "<Your workspace Key>"  
$mma = New-Object -ComObject 'AgentConfigManager.MgmtSvcCfg'  
$mma.AddCloudWorkspace($workspaceId, $workspaceKey)  
$mma.ReloadConfiguration()
```

### Add a workspace in Azure for US Government using PowerShell

```
$workspaceId = "<Your workspace Id>"  
$workspaceKey = "<Your workspace Key>"  
$mma = New-Object -ComObject 'AgentConfigManager.MgmtSvcCfg'  
$mma.AddCloudWorkspace($workspaceId, $workspaceKey, 1)  
$mma.ReloadConfiguration()
```

#### NOTE

If you've used the command line or script previously to install or configure the agent, `EnableAzureOperationalInsights` was replaced by `AddCloudWorkspace` and `RemoveCloudWorkspace`.

## Linux agent

The following steps demonstrate how to reconfigure the Linux agent if you decide to register it with a different workspace or to remove a workspace from its configuration.

1. To verify it is registered to a workspace, run the following command:

```
/opt/microsoft/omsagent/bin/omsadmin.sh -l
```

It should return a status similar to the following example:

```
Primary Workspace: <workspaceId> Status: Onboarded(OMSAgent Running)
```

It is important that the status also shows the agent is running, otherwise the following steps to reconfigure the agent will not complete successfully.

2. If it is already registered with a workspace, remove the registered workspace by running the following command. Otherwise if it is not registered, proceed to the next step.

```
/opt/microsoft/omsagent/bin/omsadmin.sh -X
```

3. To register with a different workspace, run the following command:

```
/opt/microsoft/omsagent/bin/omsadmin.sh -w <workspace id> -s <shared key> [-d <top level domain>]
```

4. To verify your changes took effect, run the following command:

```
/opt/microsoft/omsagent/bin/omsadmin.sh -l
```

It should return a status similar to the following example:

```
Primary Workspace: <workspaceId> Status: Onboarded(OMSAgent Running)
```

The agent service does not need to be restarted in order for the changes to take effect.

## Update proxy settings

Log Analytics Agent (MMA) does not use the system proxy settings. Hence, user has to pass proxy setting while installing MMA and these settings will be stored under MMA configuration(registry) on VM. To configure the agent to communicate to the service through a proxy server or [Log Analytics gateway](#) after deployment, use

one of the following methods to complete this task.

## Windows agent

### Update settings using Control Panel

1. Sign on to the computer with an account that has administrative rights.
2. Open **Control Panel**.
3. Select **Microsoft Monitoring Agent** and then click the **Proxy Settings** tab.
4. Click **Use a proxy server** and provide the URL and port number of the proxy server or gateway. If your proxy server or Log Analytics gateway requires authentication, type the username and password to authenticate and then click **OK**.

### Update settings using PowerShell

Copy the following sample PowerShell code, update it with information specific to your environment, and save it with a PS1 file name extension. Run the script on each computer that connects directly to the Log Analytics workspace in Azure Monitor.

```
param($ProxyDomainName="https://proxy.contoso.com:30443", $cred=(Get-Credential))

# First we get the Health Service configuration object. We need to determine if we
# have the right update rollup with the API we need. If not, no need to run the rest of the script.
$healthServiceSettings = New-Object -ComObject 'AgentConfigManager.MgmtSvcCfg'

$proxyMethod = $healthServiceSettings | Get-Member -Name 'SetProxyInfo'

if (!$proxyMethod)
{
    Write-Output 'Health Service proxy API not present, will not update settings.'
    return
}

Write-Output "Clearing proxy settings."
$healthServiceSettings.SetProxyInfo('', '', '')

$ProxyUserName = $cred.username

Write-Output "Setting proxy to $ProxyDomainName with proxy username $ProxyUserName."
$healthServiceSettings.SetProxyInfo($ProxyDomainName, $ProxyUserName, $cred.GetNetworkCredential().password)
```

## Linux agent

Perform the following steps if your Linux computers need to communicate through a proxy server or Log Analytics gateway. The proxy configuration value has the following syntax

[protocol://][user:password@]proxyhost[:port]. The *proxyhost* property accepts a fully qualified domain name or IP address of the proxy server.

1. Edit the file `/etc/opt/microsoft/omsagent/proxy.conf` by running the following commands and change the values to your specific settings.

```
proxyconf="https://proxyuser:proxypassword@proxyserver01:30443"
sudo echo $proxyconf >/etc/opt/microsoft/omsagent/proxy.conf
sudo chown omsagent:omiusers /etc/opt/microsoft/omsagent/proxy.conf
```

2. Restart the agent by running the following command:

```
sudo /opt/microsoft/omsagent/bin/service_control restart [<workspace id>]
```

If you see "cURL failed to perform on this base url" in the log, you can try removing '\n' in proxy.conf EOF to resolve the failure:

```
od -c /etc/opt/microsoft/omsagent/proxy.conf
cat /etc/opt/microsoft/omsagent/proxy.conf | tr -d '\n' > /etc/opt/microsoft/omsagent/proxy2.conf
rm /etc/opt/microsoft/omsagent/proxy.conf
mv /etc/opt/microsoft/omsagent/proxy2.conf /etc/opt/microsoft/omsagent/proxy.conf
sudo chown omsagent:omiusers /etc/opt/microsoft/omsagent/proxy.conf
sudo /opt/microsoft/omsagent/bin/service_control restart [<workspace id>]
```

## Uninstall agent

Use one of the following procedures to uninstall the Windows or Linux agent using the command line or setup wizard.

### Windows agent

#### Uninstall from Control Panel

1. Sign on to the computer with an account that has administrative rights.
2. In Control Panel, click **Programs and Features**.
3. In **Programs and Features**, click **Microsoft Monitoring Agent**, click **Uninstall**, and then click **Yes**.

#### NOTE

The Agent Setup Wizard can also be run by double-clicking **MMASetup-<platform>.exe**, which is available for download from a workspace in the Azure portal.

#### Uninstall from the command line

The downloaded file for the agent is a self-contained installation package created with IExpress. The setup program for the agent and supporting files are contained in the package and need to be extracted in order to properly uninstall using the command line shown in the following example.

1. Sign on to the computer with an account that has administrative rights.
2. To extract the agent installation files, from an elevated command prompt run `extract MMASetup-<platform>.exe` and it will prompt you for the path to extract files to. Alternatively, you can specify the path by passing the arguments `extract MMASetup-<platform>.exe /c:<Path> /t:<Path>`. For more information on the command-line switches supported by IExpress, see [Command-line switches for IExpress](#) and then update the example to suit your needs.
3. At the prompt, type `%WinDir%\System32\msiexec.exe /x <Path>:\MOMAgent.msi /qb`.

### Linux agent

To remove the agent, run the following command on the Linux computer. The `--purge` argument completely removes the agent and its configuration.

```
wget https://raw.githubusercontent.com/Microsoft/OMS-Agent-for-Linux/master/installer/scripts/onboard_agent.sh && sh onboard_agent.sh --purge
```

## Configure agent to report to an Operations Manager management group

### Windows agent

Perform the following steps to configure the Log Analytics agent for Windows to report to a System Center Operations Manager management group.

#### **NOTE**

As part of the ongoing transition from Microsoft Operations Management Suite to Azure Monitor, the Operations Management Suite Agent for Windows or Linux will be referred to as the Log Analytics agent for Windows and Log Analytics agent for Linux.

1. Sign on to the computer with an account that has administrative rights.
2. Open **Control Panel**.
3. Click **Microsoft Monitoring Agent** and then click the **Operations Manager** tab.
4. If your Operations Manager servers have integration with Active Directory, click **Automatically update management group assignments from AD DS**.
5. Click **Add** to open the **Add a Management Group** dialog box.
6. In **Management group name** field, type the name of your management group.
7. In the **Primary management server** field, type the computer name of the primary management server.
8. In the **Management server port** field, type the TCP port number.
9. Under **Agent Action Account**, choose either the Local System account or a local domain account.
10. Click **OK** to close the **Add a Management Group** dialog box and then click **OK** to close the **Microsoft Monitoring Agent Properties** dialog box.

#### **Linux agent**

Perform the following steps to configure the Log Analytics agent for Linux to report to a System Center Operations Manager management group.

#### **NOTE**

As part of the ongoing transition from Microsoft Operations Management Suite to Azure Monitor, the Operations Management Suite Agent for Windows or Linux will be referred to as the Log Analytics agent for Windows and Log Analytics agent for Linux.

1. Edit the file `/etc/opt/omi/conf/omiserver.conf`
2. Ensure that the line beginning with `httpsport=` defines the port 1270. Such as: `httpsport=1270`
3. Restart the OMI server: `sudo /opt/omi/bin/service_control restart`

## Next steps

- Review [Troubleshooting the Linux agent](#) if you encounter issues while installing or managing the Linux agent.
- Review [Troubleshooting the Windows agent](#) if you encounter issues while installing or managing the Windows agent.

# Agent Health solution in Azure Monitor

9/21/2022 • 5 minutes to read • [Edit Online](#)

The Agent Health solution in Azure helps you understand which monitoring agents are unresponsive and submitting operational data. That includes all the agents that report directly to the Log Analytics workspace in Azure Monitor or to a System Center Operations Manager management group connected to Azure Monitor.

You can also use the Agent Health solution to:

- Keep track of how many agents are deployed and where they're distributed geographically.
- Perform other queries to maintain awareness of the distribution of agents deployed in Azure, in other cloud environments, or on-premises.

## Prerequisites

Before you deploy this solution, confirm that you have supported [Windows agents](#) reporting to the Log Analytics workspace or reporting to an [Operations Manager management group](#) integrated with your workspace.

## Management packs

If your Operations Manager management group is connected to a Log Analytics workspace, the following management packs are installed in Operations Manager. These management packs are also installed on directly connected Windows computers after you add this solution.

- Microsoft System Center Advisor HealthAssessment Direct Channel Intelligence Pack  
(Microsoft.IntelligencePacks.HealthAssessmentDirect)
- Microsoft System Center Advisor HealthAssessment Server Channel Intelligence Pack  
(Microsoft.IntelligencePacks.HealthAssessmentViaServer).

There's nothing to configure or manage with these management packs. For more information on how solution management packs are updated, see [Connect Operations Manager to Log Analytics](#).

## Configuration

Add the Agent Health solution to your Log Analytics workspace by using the process described in [Add solutions](#). No further configuration is required.

## Supported agents

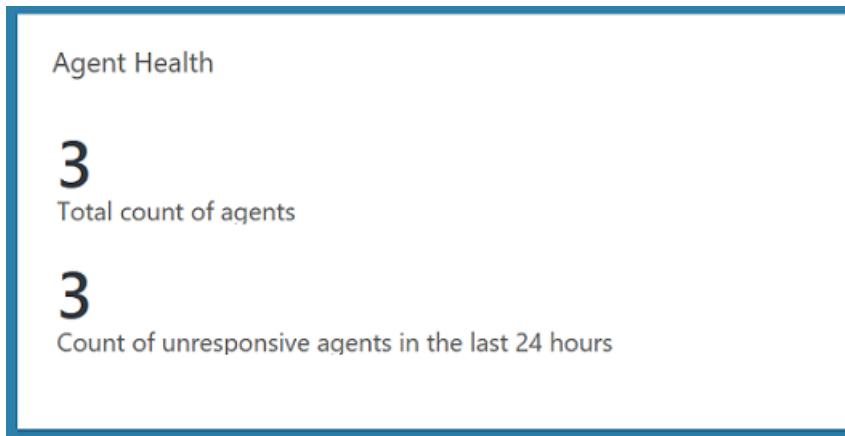
The following table describes the connected sources that this solution supports.

CONNECTED SOURCE	SUPPORTED	DESCRIPTION
Windows agents	Yes	Heartbeat events are collected from direct Windows agents.

CONNECTED SOURCE	SUPPORTED	DESCRIPTION
System Center Operations Manager management group	Yes	Heartbeat events are collected from agents that report to the management group every 60 seconds and then forwarded to Azure Monitor. A direct connection from Operations Manager agents to Azure Monitor is not required. Heartbeat event data is forwarded from the management group to the Log Analytics workspace.

## Using the solution

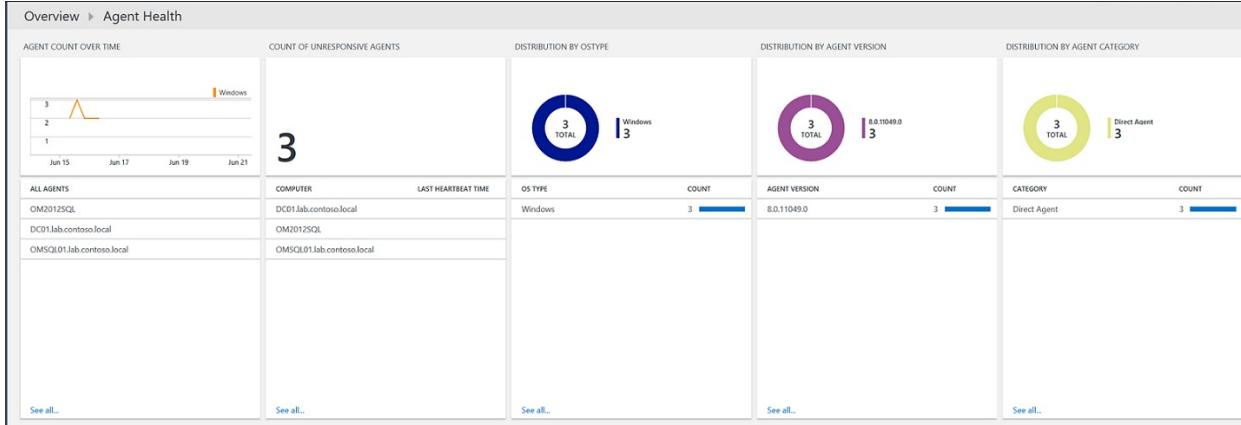
When you add the solution to your Log Analytics workspace, the **Agent Health** tile is added to your dashboard. This tile shows the total number of agents and the number of unresponsive agents in the last 24 hours.



Select the **Agent Health** tile to open the **Agent Health** dashboard. The dashboard includes the columns in the following table. Each column lists the top 10 events by count that match that column's criteria for the specified time range. You can run a log search that provides the entire list by selecting **See all** beneath each column, or by selecting the column heading.

COLUMN	DESCRIPTION
Agent count over time	A trend of your agent count over a period of seven days for both Linux and Windows agents
Count of unresponsive agents	A list of agents that haven't sent a heartbeat in the past 24 hours
Distribution by OS type	A partition of how many Windows and Linux agents you have in your environment
Distribution by agent version	A partition of the agent versions installed in your environment and a count of each one
Distribution by agent category	A partition of the categories of agents that are sending up heartbeat events: direct agents, Operations Manager agents, or the Operations Manager management server
Distribution by management group	A partition of the Operations Manager management groups in your environment

COLUMN	DESCRIPTION
Geo-location of agents	A partition of the countries/regions where you have agents, and a total count of the number of agents that have been installed in each country/region
Count of gateways installed	The number of servers that have the Log Analytics gateway installed, and a list of these servers



## Azure Monitor log records

The solution creates one type of record in the Log Analytics workspace: heartbeat. Heartbeat records have the properties in the following table.

PROPERTY	DESCRIPTION
Type	Heartbeat
Category	Direct Agent, SCOM Agent, or SCOM Management Server
Computer	Computer name
OSType	Windows or Linux operating system
OSMajorVersion	Operating system major version
OSMinorVersion	Operating system minor version
Version	Log Analytics agent or Operations Manager agent version
SCAgentChannel1	Direct and/or SCManagementServer
IsGatewayInstalled	true if the Log Analytics gateway is installed; otherwise false
ComputerIP	Public IP address for an Azure virtual machine, if one is available; Azure SNAT address (not the private IP address) for a virtual machine that uses a private IP

PROPERTY	DESCRIPTION
ComputerPrivateIPs	List of private IPs of the computer
RemoteIPCountry	Geographic location where the computer is deployed
ManagementGroupName	Name of the Operations Manager management group
SourceComputerId	Unique ID of the computer
RemoteIPLongitude	Longitude of the computer's geographic location
RemoteIPLatitude	Latitude of the computer's geographic location

Each agent that reports to an Operations Manager management server will send two heartbeats. The `SCAgentChannel1` property's value will include both `Direct` and `SCManagementServer`, depending on what data sources and monitoring solutions you've enabled in your subscription.

If you recall, data from solutions is sent either:

- Directly from an Operations Manager management server to Azure Monitor
- Directly from the agent to Azure Monitor, because of the volume of data collected on the agent

For heartbeat events that have the value `SCManagementServer`, the `ComputerIP` value is the IP address of the management server because it actually uploads the data. For heartbeats where `SCAgentChannel1` is set to `Direct`, it's the public IP address of the agent.

## Sample log searches

The following table provides sample log searches for records that the solution collects.

QUERY	DESCRIPTION
Heartbeat   distinct Computer	Total number of agents
Heartbeat   summarize LastCall = max(TimeGenerated) by Computer   where LastCall < ago(24h)	Count of unresponsive agents in the last 24 hours
Heartbeat   summarize LastCall = max(TimeGenerated) by Computer   where LastCall < ago(15m)	Count of unresponsive agents in the last 15 minutes
Heartbeat   where TimeGenerated > ago(24h) and Computer in ((Heartbeat   where TimeGenerated > ago(24h)   distinct Computer))   summarize LastCall = max(TimeGenerated) by Computer	Computers online in the last 24 hours
Heartbeat   where TimeGenerated > ago(24h) and Computer !in ((Heartbeat   where TimeGenerated > ago(30m)   distinct Computer))   summarize LastCall = max(TimeGenerated) by Computer	Total agents offline in the last 30 minutes (for the last 24 hours)
Heartbeat   summarize AggregatedValue = dcoun(Computer) by OSType	Trend of the number of agents over time by OS type

QUERY	DESCRIPTION
Heartbeat   summarize AggregatedValue = dcount(Computer) by OSType	Distribution by OS type
Heartbeat   summarize AggregatedValue = dcount(Computer) by Version	Distribution by agent version
Heartbeat   summarize AggregatedValue = count() by Category	Distribution by agent category
Heartbeat   summarize AggregatedValue = dcount(Computer) by ManagementGroupName	Distribution by management group
Heartbeat   summarize AggregatedValue = dcount(Computer) by RemoteIPCountry	Geo-location of agents
Heartbeat   where iff(isnotnull(toint(IsGatewayInstalled)), IsGatewayInstalled == true, IsGatewayInstalled == "true") == true   distinct Computer	Number of Log Analytics gateways installed

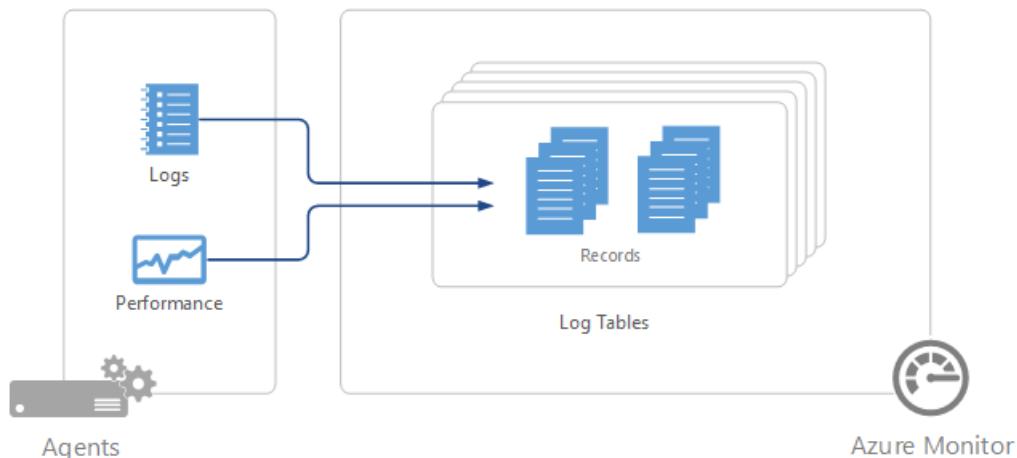
## Next steps

- Learn about [generating alerts from log queries in Azure Monitor](#).

# Log Analytics agent data sources in Azure Monitor

9/21/2022 • 3 minutes to read • [Edit Online](#)

The data that Azure Monitor collects from virtual machines with the legacy [Log Analytics](#) agent is defined by the data sources that you configure on the [Log Analytics workspace](#). Each data source creates records of a particular type with each type having its own set of properties.



## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

## IMPORTANT

The data sources described in this article apply only to virtual machines running the Log Analytics agent.

## Summary of data sources

The following table lists the agent data sources that are currently available with the Log Analytics agent. Each has a link to a separate article providing detail for that data source. It also provides information on their method and frequency of collection.

DATA SOURCE	PLATFORM	LOG ANALYTICS AGENT	OPERATIONS MANAGER AGENT	AZURE STORAGE	OPERATIONS MANAGER REQUIRED?	OPERATIONS MANAGER AGENT DATA SENT VIA MANAGEMENT GROUP	COLLECTION FREQUENCY
Custom logs	Windows	•					on arrival
Custom logs	Linux	•					on arrival

DATA SOURCE	PLATFORM	LOG ANALYTICS AGENT	OPERATIONS MANAGER AGENT	AZURE STORAGE	OPERATIONS MANAGER REQUIRED?	OPERATIONS MANAGER DATA SENT VIA MANAGEMENT GROUP	COLLECTION FREQUENCY
IIS logs	Windows	•	•	•			depends on Log File Rollover setting
Performance counters	Windows	•	•				as scheduled, minimum of 10 seconds
Performance counters	Linux	•					as scheduled, minimum of 10 seconds
Syslog	Linux	•					from Azure storage: 10 minutes; from agent: on arrival
Windows Event logs	Windows	•	•	•		•	on arrival

## Configuring data sources

To configure data sources for Log Analytics agents, go to the **Log Analytics workspaces** menu in the Azure portal and select a workspace. Click on **Agents configuration**. Select the tab for the data source you want to configure. You can follow the links in the table above to documentation for each data source and details on their configuration.

Any configuration is delivered to all agents connected to that workspace. You cannot exclude any connected agents from this configuration.

## Data collection

Data source configurations are delivered to agents that are directly connected to Azure Monitor within a few minutes. The specified data is collected from the agent and delivered directly to Azure Monitor at intervals specific to each data source. See the documentation for each data source for these specifics.

For System Center Operations Manager agents in a connected management group, data source configurations are translated into management packs and delivered to the management group every 5 minutes by default. The agent downloads the management pack like any other and collects the specified data. Depending on the data source, the data will be either sent to a management server which forwards the data to the Azure Monitor, or the agent will send the data to Azure Monitor without going through the management server. See [Data collection details for monitoring solutions in Azure](#) for details. You can read about details of connecting Operations Manager and Azure Monitor and modifying the frequency that configuration is delivered at [Configure Integration with System Center Operations Manager](#).

If the agent is unable to connect to Azure Monitor or Operations Manager, it will continue to collect data that it will deliver when it establishes a connection. Data can be lost if the amount of data reaches the maximum cache size for the client, or if the agent is not able to establish a connection within 24 hours.

## Log records

All log data collected by Azure Monitor is stored in the workspace as records. Records collected by different data sources will have their own set of properties and be identified by their **Type** property. See the documentation for each data source and solution for details on each record type.

## Next steps

- Learn about [monitoring solutions](#) that add functionality to Azure Monitor and also collect data into the workspace.
- Learn about [log queries](#) to analyze the data collected from data sources and monitoring solutions.
- Configure [alerts](#) to proactively notify you of critical data collected from data sources and monitoring solutions.

# Collecting custom JSON data sources with the Log Analytics agent for Linux in Azure Monitor

9/21/2022 • 2 minutes to read • [Edit Online](#)

## NOTE

As part of the ongoing transition from Microsoft Operations Management Suite to Azure Monitor, the Operations Management Suite Agent for Windows or Linux will be referred to as the Log Analytics agent for Windows and Log Analytics agent for Linux.

Custom JSON data sources can be collected into [Azure Monitor](#) using the Log Analytics agent for Linux. These custom data sources can be simple scripts returning JSON such as `curl` or one of [FluentD's 300+ plugins](#). This article describes the configuration required for this data collection.

## NOTE

Log Analytics agent for Linux v1.1.0-217+ is required for Custom JSON Data.

## Configuration

### Configure input plugin

To collect JSON data in Azure Monitor, add `oms.api.` to the start of a FluentD tag in an input plugin.

For example, following is a separate configuration file `exec-json.conf` in `/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/`. This uses the FluentD plugin `exec` to run a `curl` command every 30 seconds. The output from this command is collected by the JSON output plugin.

```
<source>
  type exec
  command 'curl localhost/json.output'
  format json
  tag oms.api.httpresponse
  run_interval 30s
</source>

<match oms.api.httpresponse>
  type out_oms_api
  log_level info

  buffer_chunk_limit 5m
  buffer_type file
  buffer_path /var/opt/microsoft/omsagent/<workspace id>/state/out_oms_api_httpresponse*.buffer
  buffer_queue_limit 10
  flush_interval 20s
  retry_limit 10
  retry_wait 30s
</match>
```

The configuration file added under `/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/` will require to have its ownership changed with the following command.

```
sudo chown omsagent:omiusers /etc/opt/microsoft/omsagent/conf/omsagent.d/exec-json.conf
```

## Configure output plugin

Add the following output plugin configuration to the main configuration in

```
/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf or as a separate configuration file placed in  
/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/
```

```
<match oms.api.**>  
  type out_oms_api  
  log_level info  
  
  buffer_chunk_limit 5m  
  buffer_type file  
  buffer_path /var/opt/microsoft/omsagent/<workspace id>/state/out_oms_api*.buffer  
  buffer_queue_limit 10  
  flush_interval 20s  
  retry_limit 10  
  retry_wait 30s  
</match>
```

## Restart Log Analytics agent for Linux

Restart the Log Analytics agent for Linux service with the following command.

```
sudo /opt/microsoft/omsagent/bin/service_control restart
```

## Output

The data will be collected in Azure Monitor with a record type of `<FLUENTD_TAG>_CL`.

For example, the custom tag `tag oms.api.tomcat` in Azure Monitor with a record type of `tomcat_CL`. You could retrieve all records of this type with the following log query.

```
Type=tomcat_CL
```

Nested JSON data sources are supported, but are indexed based off of parent field. For example, the following JSON data is returned from a log query as `tag_s : "[{ "a":"1", "b":"2" }]`.

```
{  
  "tag": [{  
    "a":"1",  
    "b":"2"  
  }]  
}
```

## Next steps

- Learn about [log queries](#) to analyze the data collected from data sources and solutions.

# Collect data from CollectD on Linux agents in Azure Monitor

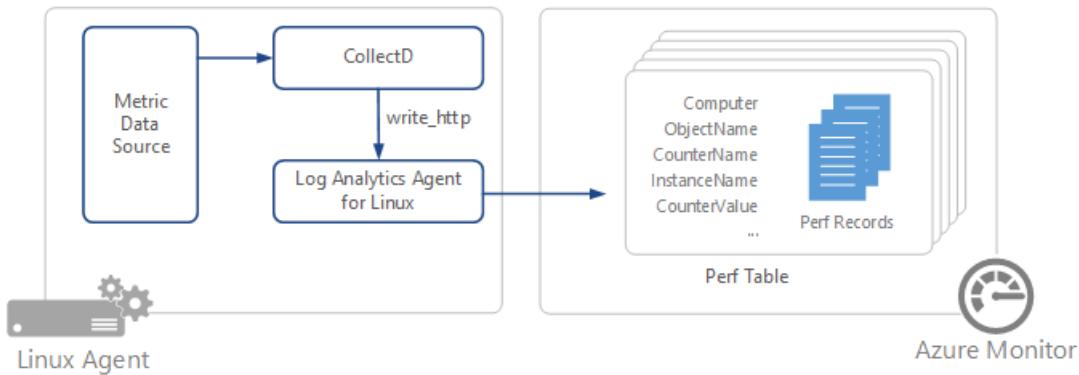
9/21/2022 • 3 minutes to read • [Edit Online](#)

CollectD is an open source Linux daemon that periodically collects performance metrics from applications and system level information. Example applications include the Java Virtual Machine (JVM), MySQL Server, and Nginx. This article provides information on collecting performance data from CollectD in Azure Monitor using the Log Analytics agent.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

A full list of available plugins can be found at [Table of Plugins](#).



The following CollectD configuration is included in the Log Analytics agent for Linux to route CollectD data to the Log Analytics agent for Linux.

## NOTE

As part of the ongoing transition from Microsoft Operations Management Suite to Azure Monitor, the Operations Management Suite Agent for Windows or Linux will be referred to as the Log Analytics agent for Windows and Log Analytics agent for Linux.

```
LoadPlugin write_http

<Plugin write_http>
  <Node "oms">
    URL "127.0.0.1:26000/oms.collectd"
    Format "JSON"
    StoreRates true
  </Node>
</Plugin>
```

Additionally, if using an versions of collectD before 5.5 use the following configuration instead.

```
LoadPlugin write_http

<Plugin write_http>
  <URL "127.0.0.1:26000/oms.collectd">
    Format "JSON"
    StoreRates true
  </URL>
</Plugin>
```

The CollectD configuration uses the default `write_http` plugin to send performance metric data over port 26000 to Log Analytics agent for Linux.

**NOTE**

This port can be configured to a custom-defined port if needed.

The Log Analytics agent for Linux also listens on port 26000 for CollectD metrics and then converts them to Azure Monitor schema metrics. The following is the Log Analytics agent for Linux configuration `collectd.conf`.

```
<source>
  type http
  port 26000
  bind 127.0.0.1
</source>

<filter oms.collectd>
  type filter_collectd
</filter>
```

**NOTE**

CollectD by default is set to read values at a 10-second `interval`. As this directly affects the volume of data sent to Azure Monitor Logs, you might need to tune this interval within the CollectD configuration to strike a good balance between the monitoring requirements and associated costs and usage for Azure Monitor Logs.

## Versions supported

- Azure Monitor currently supports CollectD version 4.8 and above.
- Log Analytics agent for Linux v1.1.0-217 or above is required for CollectD metric collection.

## Configuration

The following are basic steps to configure collection of CollectD data in Azure Monitor.

1. Configure CollectD to send data to the Log Analytics agent for Linux using the `write_http` plugin.
2. Configure the Log Analytics agent for Linux to listen for the CollectD data on the appropriate port.
3. Restart CollectD and Log Analytics agent for Linux.

### Configure CollectD to forward data

1. To route CollectD data to the Log Analytics agent for Linux, `oms.conf` needs to be added to CollectD's configuration directory. The destination of this file depends on the Linux distro of your machine.

If your CollectD config directory is located in /etc/collectd.d/:

```
sudo cp /etc/opt/microsoft/omsagent/sysconf/omsagent.d/oms.conf /etc/collectd.d/oms.conf
```

If your CollectD config directory is located in /etc/collectd/collectd.conf.d/:

```
sudo cp /etc/opt/microsoft/omsagent/sysconf/omsagent.d/oms.conf  
/etc/collectd/collectd.conf.d/oms.conf
```

**NOTE**

For CollectD versions before 5.5 you will have to modify the tags in `oms.conf` as shown above.

2. Copy collectd.conf to the desired workspace's omsagent configuration directory.

```
sudo cp /etc/opt/microsoft/omsagent/sysconf/omsagent.d/collectd.conf  
/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/  
sudo chown omsagent:omiusers /etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/collectd.conf
```

3. Restart CollectD and Log Analytics agent for Linux with the following commands.

```
sudo service collectd restart  
sudo /opt/microsoft/omsagent/bin/service_control restart
```

## CollectD metrics to Azure Monitor schema conversion

To maintain a familiar model between infrastructure metrics already collected by Log Analytics agent for Linux and the new metrics collected by CollectD the following schema mapping is used:

COLLECTD METRIC FIELD	AZURE MONITOR FIELD
<code>host</code>	Computer
<code>plugin</code>	None
<code>plugin_instance</code>	Instance Name If <code>plugin_instance</code> is <code>null</code> then <code>InstanceName=_Total</code>
<code>type</code>	ObjectName
<code>type_instance</code>	CounterName If <code>type_instance</code> is <code>null</code> then <code>CounterName=blank</code>
<code>dsnames[]</code>	CounterName
<code>dstypes</code>	None
<code>values[]</code>	CounterValue

## Next steps

- Learn about [log queries](#) to analyze the data collected from data sources and solutions.

- Use [Custom Fields](#) to parse data from syslog records into individual fields.

# Collect Syslog data sources with Log Analytics agent

9/21/2022 • 6 minutes to read • [Edit Online](#)

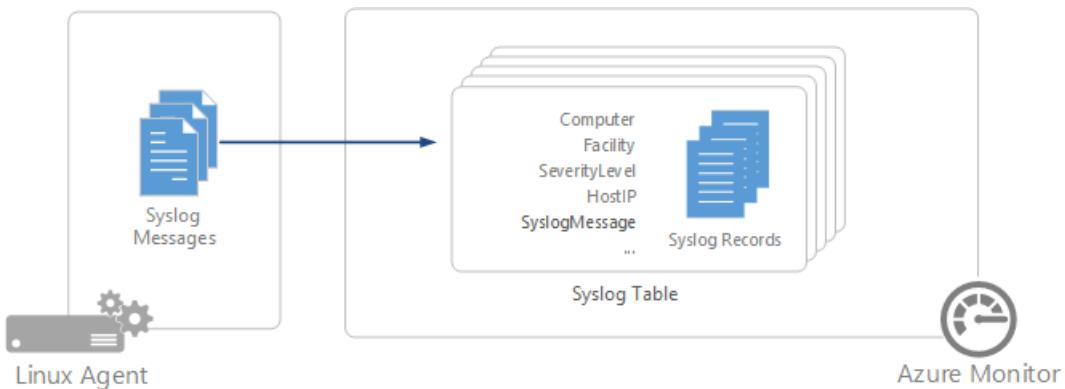
Syslog is an event logging protocol that is common to Linux. Applications will send messages that may be stored on the local machine or delivered to a Syslog collector. When the Log Analytics agent for Linux is installed, it configures the local Syslog daemon to forward messages to the agent. The agent then sends the message to Azure Monitor where a corresponding record is created.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

## NOTE

Azure Monitor supports collection of messages sent by rsyslog or syslog-ng, where rsyslog is the default daemon. The default syslog daemon on version 5 of Red Hat Enterprise Linux, CentOS, and Oracle Linux version (sysklog) is not supported for syslog event collection. To collect syslog data from this version of these distributions, the [rsyslog daemon](#) should be installed and configured to replace sysklog.



The following facilities are supported with the Syslog collector:

- kern
- user
- mail
- daemon
- auth
- syslog
- lpr
- news
- uucp
- cron
- authpriv
- ftp

- local0-local7

For any other facility, [configure a Custom Logs data source](#) in Azure Monitor.

## Configuring Syslog

The Log Analytics agent for Linux will only collect events with the facilities and severities that are specified in its configuration. You can configure Syslog through the Azure portal or by managing configuration files on your Linux agents.

### Configure Syslog in the Azure portal

Configure Syslog from the [Agent configuration menu](#) for the Log Analytics workspace. This configuration is delivered to the configuration file on each Linux agent.

You can add a new facility by clicking **Add facility**. For each facility, only messages with the selected severities will be collected. Check the severities for the particular facility that you want to collect. You cannot provide any additional criteria to filter messages.

Facility name	Emergency	Alert	Critical	Error	Warning	Notice	Info
cron	<input checked="" type="checkbox"/>						
syslog	<input checked="" type="checkbox"/>						
user	<input checked="" type="checkbox"/>						

By default, all configuration changes are automatically pushed to all agents. If you want to configure Syslog manually on each Linux agent, then uncheck the box *Apply below configuration to my machines*.

### Configure Syslog on Linux agent

When the [Log Analytics agent is installed on a Linux client](#), it installs a default syslog configuration file that defines the facility and severity of the messages that are collected. You can modify this file to change the configuration. The configuration file is different depending on the Syslog daemon that the client has installed.

#### NOTE

If you edit the syslog configuration, you must restart the syslog daemon for the changes to take effect.

#### rsyslog

The configuration file for rsyslog is located at `/etc/rsyslog.d/95-omsagent.conf`. Its default contents are shown below. This collects syslog messages sent from the local agent for all facilities with a level of warning or higher.

```
kern.warning      @127.0.0.1:25224
user.warning     @127.0.0.1:25224
daemon.warning   @127.0.0.1:25224
auth.warning     @127.0.0.1:25224
syslog.warning   @127.0.0.1:25224
uucp.warning     @127.0.0.1:25224
authpriv.warning @127.0.0.1:25224
ftp.warning      @127.0.0.1:25224
cron.warning     @127.0.0.1:25224
local0.warning   @127.0.0.1:25224
local1.warning   @127.0.0.1:25224
local2.warning   @127.0.0.1:25224
local3.warning   @127.0.0.1:25224
local4.warning   @127.0.0.1:25224
local5.warning   @127.0.0.1:25224
local6.warning   @127.0.0.1:25224
local7.warning   @127.0.0.1:25224
```

You can remove a facility by removing its section of the configuration file. You can limit the severities that are collected for a particular facility by modifying that facility's entry. For example, to limit the user facility to messages with a severity of error or higher you would modify that line of the configuration file to the following:

```
user.error      @127.0.0.1:25224
```

### **syslog-ng**

The configuration file for syslog-ng is located at **/etc/syslog-ng/syslog-ng.conf**. Its default contents are shown below. This collects syslog messages sent from the local agent for all facilities and all severities.

```

#
# Warnings (except iptables) in one file:
#
destination warn { file("/var/log/warn" fsync(yes)); };
log { source(src); filter(f_warn); destination(warn); };

#OMS_Destination
destination d_oms { udp("127.0.0.1" port(25224)); };

#OMS_facility = auth
filter f_auth_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(auth); };
log { source(src); filter(f_auth_oms); destination(d_oms); };

#OMS_facility = authpriv
filter f_authpriv_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(authpriv); };
log { source(src); filter(f_authpriv_oms); destination(d_oms); };

#OMS_facility = cron
filter f_cron_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(cron); };
log { source(src); filter(f_cron_oms); destination(d_oms); };

#OMS_facility = daemon
filter f_daemon_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(daemon); };
log { source(src); filter(f_daemon_oms); destination(d_oms); };

#OMS_facility = kern
filter f_kern_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(kern); };
log { source(src); filter(f_kern_oms); destination(d_oms); };

#OMS_facility = local0
filter f_local0_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(local0); };
log { source(src); filter(f_local0_oms); destination(d_oms); };

#OMS_facility = local1
filter f_local1_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(local1); };
log { source(src); filter(f_local1_oms); destination(d_oms); };

#OMS_facility = mail
filter f_mail_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(mail); };
log { source(src); filter(f_mail_oms); destination(d_oms); };

#OMS_facility = syslog
filter f_syslog_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(syslog); };
log { source(src); filter(f_syslog_oms); destination(d_oms); };

#OMS_facility = user
filter f_user_oms { level(alert,crit,debug,emerg,err,info,notice,warning) and facility(user); };
log { source(src); filter(f_user_oms); destination(d_oms); };

```

You can remove a facility by removing its section of the configuration file. You can limit the severities that are collected for a particular facility by removing them from its list. For example, to limit the user facility to just alert and critical messages, you would modify that section of the configuration file to the following:

```

#OMS_facility = user
filter f_user_oms { level(alert,crit) and facility(user); };
log { source(src); filter(f_user_oms); destination(d_oms); };

```

## Collecting data from additional Syslog ports

The Log Analytics agent listens for Syslog messages on the local client on port 25224. When the agent is installed, a default syslog configuration is applied and found in the following location:

- Rsyslog: /etc/rsyslog.d/95-omsagent.conf

- Syslog-**ng**: `/etc/syslog-ng/syslog-ng.conf`

You can change the port number by creating two configuration files: a FluentD config file and a rsyslog-or-syslog-**ng** file depending on the Syslog daemon you have installed.

- The FluentD config file should be a new file located in: `/etc/opt/microsoft/omsagent/conf/omsagent.d` and replace the value in the **port** entry with your custom port number.

```
<source>
  type syslog
  port %SYSLOG_PORT%
  bind 127.0.0.1
  protocol_type udp
  tag oms.syslog
</source>
<filter oms.syslog.**>
  type filter_syslog
```

- For rsyslog, you should create a new configuration file located in: `/etc/rsyslog.d/` and replace the value `%SYSLOG_PORT%` with your custom port number.

#### **NOTE**

If you modify this value in the configuration file `95-omsagent.conf`, it will be overwritten when the agent applies a default configuration.

```
# OMS Syslog collection for workspace %WORKSPACE_ID%
kern.warning      @127.0.0.1:%SYSLOG_PORT%
user.warning      @127.0.0.1:%SYSLOG_PORT%
daemon.warning    @127.0.0.1:%SYSLOG_PORT%
auth.warning      @127.0.0.1:%SYSLOG_PORT%
```

- The syslog-**ng** config should be modified by copying the example configuration shown below and adding the custom modified settings to the end of the `syslog-ng.conf` configuration file located in `/etc/syslog-ng/`. Do **not** use the default label `%WORKSPACE_ID%_oms` or `%WORKSPACE_ID_OMS`, define a custom label to help distinguish your changes.

#### **NOTE**

If you modify the default values in the configuration file, they will be overwritten when the agent applies a default configuration.

```
filter f_custom_filter { level(warning) and facility(auth); };
destination d_custom_dest { udp("127.0.0.1" port(%SYSLOG_PORT%)); };
log { source(s_src); filter(f_custom_filter); destination(d_custom_dest); };
```

After completing the changes, the Syslog and the Log Analytics agent service needs to be restarted to ensure the configuration changes take effect.

## Syslog record properties

Syslog records have a type of **Syslog** and have the properties in the following table.

PROPERTY	DESCRIPTION
Computer	Computer that the event was collected from.
Facility	Defines the part of the system that generated the message.
HostIP	IP address of the system sending the message.
HostName	Name of the system sending the message.
SeverityLevel	Severity level of the event.
SyslogMessage	Text of the message.
ProcessID	ID of the process that generated the message.
EventTime	Date and time that the event was generated.

## Log queries with Syslog records

The following table provides different examples of log queries that retrieve Syslog records.

QUERY	DESCRIPTION
Syslog	All Syslogs.
Syslog   where SeverityLevel == "error"	All Syslog records with severity of error.
Syslog   summarize AggregatedValue = count() by Computer	Count of Syslog records by computer.
Syslog   summarize AggregatedValue = count() by Facility	Count of Syslog records by facility.

## Next steps

- Learn about [log queries](#) to analyze the data collected from data sources and solutions.
- Use [Custom Fields](#) to parse data from syslog records into individual fields.
- [Configure Linux agents](#) to collect other types of data.

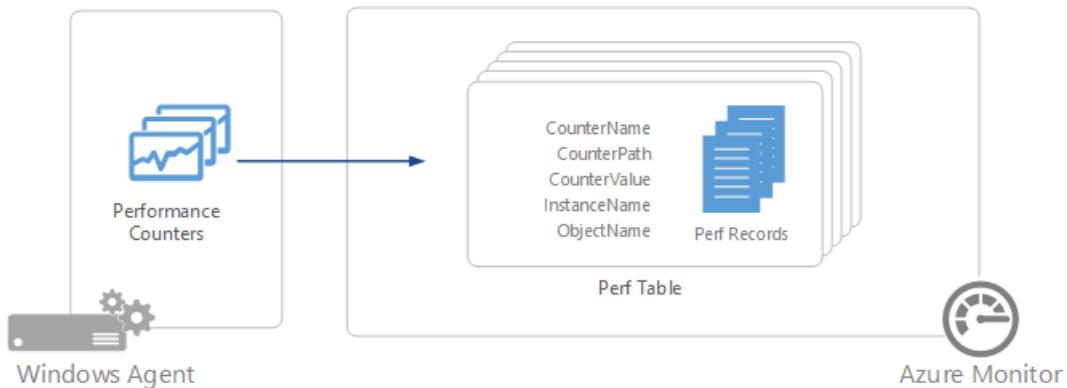
# Collect Windows and Linux performance data sources with Log Analytics agent

9/21/2022 • 7 minutes to read • [Edit Online](#)

Performance counters in Windows and Linux provide insight into the performance of hardware components, operating systems, and applications. Azure Monitor can collect performance counters from Log Analytics agents at frequent intervals for Near Real Time (NRT) analysis in addition to aggregating performance data for longer term analysis and reporting.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.



## Configuring Performance counters

Configure Performance counters from the [Agents configuration menu](#) for the Log Analytics workspace.

When you first configure Windows or Linux Performance counters for a new workspace, you are given the option to quickly create several common counters. They are listed with a checkbox next to each. Ensure that any counters you want to initially create are checked and then click **Add the selected performance counters**.

For Windows performance counters, you can choose a specific instance for each performance counter. For Linux performance counters, the instance of each counter that you choose applies to all child counters of the parent counter. The following table shows the common instances available to both Linux and Windows performance counters.

INSTANCE NAME	DESCRIPTION
_Total	Total of all the instances
*	All instances
(/ /var)	Matches instances named: / or /var

### Windows performance counters

The screenshot shows the 'Windows performance counters' section of the Agents configuration page. It includes a search bar, tabs for Windows event logs, Windows performance counters (selected), Linux performance counters, Syslog, and IIS Logs. A descriptive text explains collecting performance counters from agents at custom intervals. Below is a table for adding performance counters:

Performance counter name	Sample rate (seconds)
LogicalDisk(*)\% Free Space	10
LogicalDisk(*)\Avg. Disk sec/Read	10
LogicalDisk(*)\Avg. Disk sec/Write	10
LogicalDisk(*)\Current Disk Queue Length	10
LogicalDisk(*)\Disk Reads/sec	10

Follow this procedure to add a new Windows performance counter to collect. Please note that V2 Windows Performance Counters are not supported.

1. Click **Add performance counter**.
2. Type the name of the counter in the text box in the format *object(instance)|counter*. When you start typing, you are presented with a matching list of common counters. You can either select a counter from the list or type in one of your own. You can also return all instances for a particular counter by specifying *object|counter*.  
When collecting SQL Server performance counters from named instances, all named instance counters start with *MSSQL\$* and followed by the name of the instance. For example, to collect the Log Cache Hit Ratio counter for all databases from the Database performance object for named SQL instance INST2, specify `MSSQL$INST2:Databases(*)\Log Cache Hit Ratio`.
3. When you add a counter, it uses the default of 10 seconds for its **Sample Interval**. You can change this to a higher value of up to 1800 seconds (30 minutes) if you want to reduce the storage requirements of the collected performance data.
4. When you're done adding counters, click the **Apply** button at the top of the screen to save the configuration.

## Linux performance counters

The screenshot shows the 'Linux performance counters' section of the Agents configuration page. It includes a search bar, tabs for Windows event logs, Windows performance counters, Linux performance counters (selected), Syslog, and IIS Logs. A descriptive text explains collecting performance counters from agents at custom intervals. Below is a table for adding performance counters:

Performance counter name	Instance	Sample rate (seconds)
Logical Disk	*	10
% Used Inodes		
% Used Space		
Disk Reads/sec		
Disk Transfers/sec		

Follow this procedure to add a new Linux performance counter to collect.

1. Click **Add performance counter**.
2. Type the name of the counter in the text box in the format *object(instance)|counter*. When you start typing,

you are presented with a matching list of common counters. You can either select a counter from the list or type in one of your own.

3. All counters for an object use the same **Sample Interval**. The default is 10 seconds. You change this to a higher value of up to 1800 seconds (30 minutes) if you want to reduce the storage requirements of the collected performance data.
4. When you're done adding counters, click the **Apply** button at the top of the screen to save the configuration.

#### Configure Linux performance counters in configuration file

Instead of configuring Linux performance counters using the Azure portal, you have the option of editing configuration files on the Linux agent. Performance metrics to collect are controlled by the configuration in `/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf`.

Each object, or category, of performance metrics to collect should be defined in the configuration file as a single `<source>` element. The syntax follows the pattern below.

```
<source>
  type oms_omi
  object_name "Processor"
  instance_regex ".*"
  counter_name_regex ".*"
  interval 30s
</source>
```

The parameters in this element are described in the following table.

PARAMETERS	DESCRIPTION
object_name	Object name for the collection.
instance_regex	A <i>regular expression</i> defining which instances to collect. The value: <code>.*</code> specifies all instances. To collect processor metrics for only the <code>_Total</code> instance, you could specify <code>_Total</code> . To collect process metrics for only the <code>crond</code> or <code>sshd</code> instances, you could specify: <code>(crond\ sshd)</code> .
counter_name_regex	A <i>regular expression</i> defining which counters (for the object) to collect. To collect all counters for the object, specify: <code>.*</code> . To collect only swap space counters for the memory object, for example, you could specify: <code>.+Swap.+</code>
interval	Frequency at which the object's counters are collected.

The following table lists the objects and counters that you can specify in the configuration file. There are additional counters available for certain applications as described in [Collect performance counters for Linux applications in Azure Monitor](#).

OBJECT NAME	COUNTER NAME
Logical Disk	% Free Inodes
Logical Disk	% Free Space
Logical Disk	% Used Inodes
Logical Disk	% Used Space

OBJECT NAME	COUNTER NAME
Logical Disk	Disk Read Bytes/sec
Logical Disk	Disk Reads/sec
Logical Disk	Disk Transfers/sec
Logical Disk	Disk Write Bytes/sec
Logical Disk	Disk Writes/sec
Logical Disk	Free Megabytes
Logical Disk	Logical Disk Bytes/sec
Memory	% Available Memory
Memory	% Available Swap Space
Memory	% Used Memory
Memory	% Used Swap Space
Memory	Available MBytes Memory
Memory	Available MBytes Swap
Memory	Page Reads/sec
Memory	Page Writes/sec
Memory	Pages/sec
Memory	Used MBytes Swap Space
Memory	Used Memory MBytes
Network	Total Bytes Transmitted
Network	Total Bytes Received
Network	Total Bytes
Network	Total Packets Transmitted
Network	Total Packets Received
Network	Total Rx Errors
Network	Total Tx Errors

OBJECT NAME	COUNTER NAME
Network	Total Collisions
Physical Disk	Avg. Disk sec/Read
Physical Disk	Avg. Disk sec/Transfer
Physical Disk	Avg. Disk sec/Write
Physical Disk	Physical Disk Bytes/sec
Process	Pct Privileged Time
Process	Pct User Time
Process	Used Memory kBytes
Process	Virtual Shared Memory
Processor	% DPC Time
Processor	% Idle Time
Processor	% Interrupt Time
Processor	% IO Wait Time
Processor	% Nice Time
Processor	% Privileged Time
Processor	% Processor Time
Processor	% User Time
System	Free Physical Memory
System	Free Space in Paging Files
System	Free Virtual Memory
System	Processes
System	Size Stored In Paging Files
System	Uptime
System	Users

Following is the default configuration for performance metrics.

```

<source>
  type oms_omi
  object_name "Physical Disk"
  instance_regex ".*"
  counter_name_regex ".*"
  interval 5m
</source>

<source>
  type oms_omi
  object_name "Logical Disk"
  instance_regex ".*"
  counter_name_regex ".*"
  interval 5m
</source>

<source>
  type oms_omi
  object_name "Processor"
  instance_regex ".*"
  counter_name_regex ".*"
  interval 30s
</source>

<source>
  type oms_omi
  object_name "Memory"
  instance_regex ".*"
  counter_name_regex ".*"
  interval 30s
</source>

```

## Data collection

Azure Monitor collects all specified performance counters at their specified sample interval on all agents that have that counter installed. The data is not aggregated, and the raw data is available in all log query views for the duration specified by your log analytics workspace.

## Performance record properties

Performance records have a type of **Perf** and have the properties in the following table.

PROPERTY	DESCRIPTION
Computer	Computer that the event was collected from.
CounterName	Name of the performance counter
CounterPath	Full path of the counter in the form \\<Computer>\object(instance)\counter.
CounterValue	Numeric value of the counter.
InstanceName	Name of the event instance. Empty if no instance.
ObjectName	Name of the performance object

PROPERTY	DESCRIPTION
SourceSystem	Type of agent the data was collected from.  OpsManager – Windows agent, either direct connect or SCOM Linux – All Linux agents AzureStorage – Azure Diagnostics
TimeGenerated	Date and time the data was sampled.

## Sizing estimates

A rough estimate for collection of a particular counter at 10-second intervals is about 1 MB per day per instance. You can estimate the storage requirements of a particular counter with the following formula.

1 MB x (number of counters) x (number of agents) x (number of instances)
--

## Log queries with Performance records

The following table provides different examples of log queries that retrieve Performance records.

QUERY	DESCRIPTION
Perf	All Performance data
Perf   where Computer == "MyComputer"	All Performance data from a particular computer
Perf   where CounterName == "Current Disk Queue Length"	All Performance data for a particular counter
Perf   where ObjectName == "Processor" and CounterName == "% Processor Time" and InstanceName == "_Total"   summarize AVGCPU = avg(CounterValue) by Computer	Average CPU Utilization across all computers
Perf   where CounterName == "% Processor Time"   summarize AggregatedValue = max(CounterValue) by Computer	Maximum CPU Utilization across all computers
Perf   where ObjectName == "LogicalDisk" and CounterName == "Current Disk Queue Length" and Computer == "MyComputerName"   summarize AggregatedValue = avg(CounterValue) by InstanceName	Average Current Disk Queue length across all the instances of a given computer
Perf   where CounterName == "Disk Transfers/sec"   summarize AggregatedValue = percentile(CounterValue, 95) by Computer	95th Percentile of Disk Transfers/Sec across all computers
Perf   where CounterName == "% Processor Time" and InstanceName == "_Total"   summarize AggregatedValue = avg(CounterValue) by bin(TimeGenerated, 1h), Computer	Hourly average of CPU usage across all computers
Perf   where Computer == "MyComputer" and CounterName startswith_cs "%" and InstanceName == "_Total"   summarize AggregatedValue = percentile(CounterValue, 70) by bin(TimeGenerated, 1h), CounterName	Hourly 70 percentile of every % percent counter for a particular computer

QUERY	DESCRIPTION
<pre>Perf   where CounterName == "% Processor Time" and InstanceName == "_Total" and Computer == "MyComputer"   summarize ["min(CounterValue)"] = min(CounterValue), ["avg(CounterValue)"] = avg(CounterValue), ["percentile75(CounterValue)"] = percentile(CounterValue, 75), ["max(CounterValue)"] = max(CounterValue) by bin(TimeGenerated, 1h), Computer</pre>	Hourly average, minimum, maximum, and 75-percentile CPU usage for a specific computer
<pre>Perf   where ObjectName == "MSSQL\$INST2:Databases" and InstanceName == "master"</pre>	All Performance data from the Database performance object for the master database from the named SQL Server instance INST2.

## Next steps

- [Collect performance counters from Linux applications](#) including MySQL and Apache HTTP Server.
- Learn about [log queries](#) to analyze the data collected from data sources and solutions.
- Export collected data to [Power BI](#) for additional visualizations and analysis.

# Collect performance counters for Linux applications in Azure Monitor

9/21/2022 • 6 minutes to read • [Edit Online](#)

This article provides details for configuring the [Log Analytics agent for Linux](#) to collect performance counters for specific applications into Azure Monitor. The applications included in this article are:

- [MySQL](#)
- [Apache HTTP Server](#)

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.

## MySQL

If MySQL Server or MariaDB Server is detected on the computer when the Log Analytics agent is installed, a performance monitoring provider for MySQL Server will be automatically installed. This provider connects to the local MySQL/MariaDB server to expose performance statistics. MySQL user credentials must be configured so that the provider can access the MySQL Server.

### Configure MySQL credentials

The MySQL OMI provider requires a preconfigured MySQL user and installed MySQL client libraries in order to query the performance and health information from the MySQL instance. These credentials are stored in an authentication file that's stored on the Linux agent. The authentication file specifies what bind-address and port the MySQL instance is listening on and what credentials to use to gather metrics.

During installation of the Log Analytics agent for Linux the MySQL OMI provider will scan MySQL my.cnf configuration files (default locations) for bind-address and port and partially set the MySQL OMI authentication file.

The MySQL authentication file is stored at `/var/opt/microsoft/mysql-cimprov/auth/omsagent/mysql-auth`.

### Authentication file format

Following is the format for the MySQL OMI authentication file

```
[Port]=[Bind-Address], [username], [Base64 encoded Password]
(Port)=(Bind-Address), (username), (Base64 encoded Password)
(Port)=(Bind-Address), (username), (Base64 encoded Password)
AutoUpdate=[true|false]
```

The entries in the authentication file are described in the following table.

PROPERTY	DESCRIPTION
Port	Represents the current port the MySQL instance is listening on. Port 0 specifies that the properties following are used for default instance.

PROPERTY	DESCRIPTION
Bind-Address	Current MySQL bind-address.
username	MySQL user used to use to monitor the MySQL server instance.
Base64 encoded Password	Password of the MySQL monitoring user encoded in Base64.
AutoUpdate	Specifies whether to rescan for changes in the my.cnf file and overwrite the MySQL OMI Authentication file when the MySQL OMI Provider is upgraded.

## Default instance

The MySQL OMI authentication file can define a default instance and port number to make managing multiple MySQL instances on one Linux host easier. The default instance is denoted by an instance with port 0. All additional instances will inherit properties set from the default instance unless they specify different values. For example, if MySQL instance listening on port '3308' is added, the default instance's bind-address, username, and Base64 encoded password will be used to try and monitor the instance listening on 3308. If the instance on 3308 is bound to another address and uses the same MySQL username and password pair only the bind-address is needed, and the other properties will be inherited.

The following table has example instance settings

DESCRIPTION	FILE
Default instance and instance with port 3308.	<pre>0=127.0.0.1, myuser, cnBwdA== 3308=, , AutoUpdate=true</pre>
Default instance and instance with port 3308 and different user name and password.	<pre>0=127.0.0.1, myuser, cnBwdA== 3308=127.0.1.1, myuser2,cGluaGVhZA== AutoUpdate=true</pre>

## MySQL OMI Authentication File Program

Included with the installation of the MySQL OMI provider is a MySQL OMI authentication file program which can be used to edit the MySQL OMI Authentication file. The authentication file program can be found at the following location.

```
/opt/microsoft/mysql-cimprov/bin/mycimprovauth
```

### NOTE

The credentials file must be readable by the omsagent account. Running the mycimprovauth command as omsgent is recommended.

The following table provides details on the syntax for using mycimprovauth.

OPERATION	EXAMPLE	DESCRIPTION
autoupdate <i>false or true</i>	mycimprovauth autoupdate false	Sets whether or not the authentication file will be automatically updated on restart or update.

OPERATION	EXAMPLE	DESCRIPTION
default <i>bind-address username password</i>	mycimprovauth default 127.0.0.1 root pwd	Sets the default instance in the MySQL OMI authentication file. The password field should be entered in plain text - the password in the MySQL OMI authentication file will be Base 64 encoded.
delete <i>default or port_num</i>	mycimprovauth 3308	Deletes the specified instance by either default or by port number.
help	mycimprov help	Prints out a list of commands to use.
print	mycimprov print	Prints out an easy to read MySQL OMI authentication file.
update <i>port_num bind-address username password</i>	mycimprov update 3307 127.0.0.1 root pwd	Updates the specified instance or adds the instance if it does not exist.

The following example commands define a default user account for the MySQL server on localhost. The password field should be entered in plain text - the password in the MySQL OMI authentication file will be Base 64 encoded

```
sudo su omsagent -c '/opt/microsoft/mysql-cimprov/bin/mycimprovauth default 127.0.0.1 <username> <password>'  
sudo /opt/omi/bin/service_control restart
```

### Database Permissions Required for MySQL Performance Counters

The MySQL User requires access to the following queries to collect MySQL Server performance data.

```
SHOW GLOBAL STATUS;  
SHOW GLOBAL VARIABLES;
```

The MySQL user also requires SELECT access to the following default tables.

- information\_schema
- mysql.

These privileges can be granted by running the following grant commands.

```
GRANT SELECT ON information_schema.* TO 'monuser'@'localhost';  
GRANT SELECT ON mysql.* TO 'monuser'@'localhost';
```

#### NOTE

To grant permissions to a MySQL monitoring user the granting user must have the 'GRANT option' privilege as well as the privilege being granted.

### Define performance counters

Once you configure the Log Analytics agent for Linux to send data to Azure Monitor, you must configure the performance counters to collect. Use the procedure in [Windows and Linux performance data sources in Azure Monitor](#) with the counters in the following table.

OBJECT NAME	COUNTER NAME
MySQL Database	Disk Space in Bytes
MySQL Database	Tables
MySQL Server	Aborted Connection Pct
MySQL Server	Connection Use Pct
MySQL Server	Disk Space Use in Bytes
MySQL Server	Full Table Scan Pct
MySQL Server	InnoDB Buffer Pool Hit Pct
MySQL Server	InnoDB Buffer Pool Use Pct
MySQL Server	InnoDB Buffer Pool Use Pct
MySQL Server	Key Cache Hit Pct
MySQL Server	Key Cache Use Pct
MySQL Server	Key Cache Write Pct
MySQL Server	Query Cache Hit Pct
MySQL Server	Query Cache Prunes Pct
MySQL Server	Query Cache Use Pct
MySQL Server	Table Cache Hit Pct
MySQL Server	Table Cache Use Pct
MySQL Server	Table Lock Contention Pct

## Apache HTTP Server

If Apache HTTP Server is detected on the computer when the omsagent bundle is installed, a performance monitoring provider for Apache HTTP Server will be automatically installed. This provider relies on an Apache module that must be loaded into the Apache HTTP Server in order to access performance data. The module can be loaded with the following command:

```
sudo /opt/microsoft/apache-cimprov/bin/apache_config.sh -c
```

To unload the Apache monitoring module, run the following command:

```
sudo /opt/microsoft/apache-cimprov/bin/apache_config.sh -u
```

## Define performance counters

Once you configure the Log Analytics agent for Linux to send data to Azure Monitor, you must configure the performance counters to collect. Use the procedure in [Windows and Linux performance data sources in Azure Monitor](#) with the counters in the following table.

OBJECT NAME	COUNTER NAME
Apache HTTP Server	Busy Workers
Apache HTTP Server	Idle Workers
Apache HTTP Server	Pct Busy Workers
Apache HTTP Server	Total Pct CPU
Apache Virtual Host	Errors per Minute - Client
Apache Virtual Host	Errors per Minute - Server
Apache Virtual Host	KB per Request
Apache Virtual Host	Requests KB per Second
Apache Virtual Host	Requests per Second

## Next steps

- [Collect performance counters](#) from Linux agents.
- Learn about [log queries](#) to analyze the data collected from data sources and solutions.

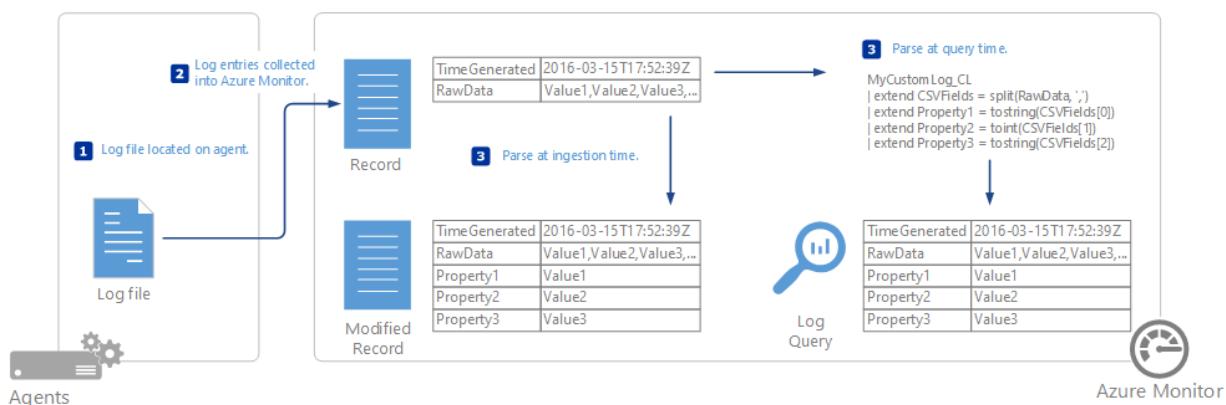
# Collect text logs with the Log Analytics agent in Azure Monitor

9/21/2022 • 8 minutes to read • [Edit Online](#)

The Custom Logs data source for the Log Analytics agent in Azure Monitor allows you to collect events from text files on both Windows and Linux computers. Many applications log information to text files instead of standard logging services, such as Windows Event log or Syslog. After the data is collected, you can either parse it into individual fields in your queries or extract it during collection to individual fields.

## IMPORTANT

The legacy [Log Analytics agent](#) will be deprecated by August 2024. Migrate to [Azure Monitor agent](#) before August 2024 to continue ingesting data.



The log files to be collected must match the following criteria:

- The log must either have a single entry per line or use a timestamp matching one of the following formats at the start of each entry:

YYYY-MM-DD HH:MM:SS  
M/D/YYYY HH:MM:SS AM/PM  
Mon DD, YYYY HH:MM:SS  
yyMMdd HH:mm:ss  
ddMMyy HH:mm:ss  
MMM d hh:mm:ss  
dd/MMM/yyyy:HH:mm:ss zzz  
yyyy-MM-ddTHH:mm:ssK

- The log file must not allow circular logging. This behavior is log rotation where the file is overwritten with new entries or the file is renamed and the same file name is reused for continued logging.
- The log file must use ASCII or UTF-8 encoding. Other formats such as UTF-16 aren't supported.
- For Linux, time zone conversion isn't supported for time stamps in the logs.
- As a best practice, the log file should include the date and time that it was created to prevent log rotation overwriting or renaming.

#### **NOTE**

If there are duplicate entries in the log file, Azure Monitor will collect them. The query results that are generated will be inconsistent. The filter results will show more events than the result count. You must validate the log to determine if the application that creates it is causing this behavior. Address the issue, if possible, before you create the custom log collection definition.

A Log Analytics workspace supports the following limits:

- Only 500 custom logs can be created.
- A table only supports up to 500 columns.
- The maximum number of characters for the column name is 500.

#### **IMPORTANT**

Custom log collection requires that the application writing the log file flushes the log content to the disk periodically. This is because the custom log collection relies on filesystem change notifications for the log file being tracked.

## Define a custom log

Use the following procedure to define a custom log file. Scroll to the end of this article for a walkthrough of a sample of adding a custom log.

### Open the Custom Log wizard

The Custom Log wizard runs in the Azure portal and allows you to define a new custom log to collect.

1. In the Azure portal, select **Log Analytics workspaces** > your workspace > **Settings**.
2. Select **Custom logs**.
3. By default, all configuration changes are automatically pushed to all agents. For Linux agents, a configuration file is sent to the Fluentd data collector.
4. Select **Add** to open the Custom Log wizard.

### Upload and parse a sample log

To start, upload a sample of the custom log. The wizard will parse and display the entries in this file for you to validate. Azure Monitor will use the delimiter that you specify to identify each record.

**New Line** is the default delimiter and is used for log files that have a single entry per line. If the line starts with a date and time in one of the available formats, you can specify a **Timestamp** delimiter, which supports entries that span more than one line.

If a timestamp delimiter is used, the **TimeGenerated** property of each record stored in Azure Monitor will be populated with the date and time specified for that entry in the log file. If a new line delimiter is used, **TimeGenerated** is populated with the date and time when Azure Monitor collected the entry.

1. Select **Browse** and browse to a sample file. This button might be labeled **Choose File** in some browsers.
2. Select **Next**.
3. The Custom Log wizard uploads the file and lists the records that it identifies.
4. Change the delimiter that's used to identify a new record. Select the delimiter that best identifies the records in your log file.
5. Select **Next**.

### Add log collection paths

You must define one or more paths on the agent where it can locate the custom log. You can either provide a

specific path and name for the log file or you can specify a path with a wildcard for the name. This step supports applications that create a new file each day or when one file reaches a certain size. You can also provide multiple paths for a single log file.

For example, an application might create a log file each day with the date included in the name as in `log20100316.txt`. A pattern for such a log might be `/log*.txt`, which would apply to any log file following the application's naming scheme.

The following table provides examples of valid patterns to specify different log files.

DESCRIPTION	PATH
All files in <code>C:\Logs</code> with .txt extension on the Windows agent	<code>C:\Logs\*.txt</code>
All files in <code>C:\Logs</code> with a name starting with <code>log</code> and a .txt extension on the Windows agent	<code>C:\Logs\log*.txt</code>
All files in <code>/var/log/audit</code> with .txt extension on the Linux agent	<code>/var/log/audit/*.txt</code>
All files in <code>/var/log/audit</code> with a name starting with <code>log</code> and a .txt extension on the Linux agent	<code>/var/log/audit/log*.txt</code>

1. Select Windows or Linux to specify which path format you're adding.
2. Enter the path and select the **+** button.
3. Repeat the process for any more paths.

### Provide a name and description for the log

The name that you specify will be used for the log type as described. It will always end with `_CL` to distinguish it as a custom log.

1. Enter a name for the log. The `_CL` suffix is automatically provided.
2. Add an optional **Description**.
3. Select **Next** to save the custom log definition.

### Validate that the custom logs are being collected

It might take up to an hour for the initial data from a new custom log to appear in Azure Monitor. Azure Monitor will start collecting entries from the logs found in the path you specified from the point that you defined the custom log. It won't retain the entries that you uploaded during the custom log creation. It will collect already existing entries in the log files that it locates.

After Azure Monitor starts collecting from the custom log, its records will be available with a log query. Use the name that you gave the custom log as the **Type** in your query.

#### NOTE

If the `RawData` property is missing from the query, you might need to close and reopen your browser.

### Parse the custom log entries

The entire log entry will be stored in a single property called **RawData**. You'll most likely want to separate the different pieces of information in each entry into individual properties for each record. For options on parsing `RawData` into multiple properties, see [Parse text data in Azure Monitor](#).

## Remove a custom log

Use the following process in the Azure portal to remove a custom log that you previously defined.

1. From the **Data** menu in the **Advanced Settings** for your workspace, select **Custom Logs** to list all your custom logs.
2. Select **Remove** next to the custom log to remove the log.

## Data collection

Azure Monitor collects new entries from each custom log approximately every 5 minutes. The agent records its place in each log file that it collects from. If the agent goes offline for a period of time, Azure Monitor collects entries from where it last left off, even if those entries were created while the agent was offline.

The entire contents of the log entry are written to a single property called **RawData**. For methods to parse each imported log entry into multiple properties, see [Parse text data in Azure Monitor](#).

## Custom log record properties

Custom log records have a type with the log name that you provide and the properties in the following table.

PROPERTY	DESCRIPTION
TimeGenerated	Date and time that the record was collected by Azure Monitor. If the log uses a time-based delimiter, this is the time collected from the entry.
SourceSystem	Type of agent the record was collected from. OpsManager – Windows agent, either direct connect or System Center Operations Manager Linux – All Linux agents
RawData	Full text of the collected entry. You'll most likely want to <a href="#">parse this data into individual properties</a> .
ManagementGroupName	Name of the management group for System Center Operations Manager agents. For other agents, this name is AOI-<workspace ID>.

## Sample walkthrough of adding a custom log

The following section walks through an example of creating a custom log. The sample log being collected has a single entry on each line starting with a date and time and then comma-delimited fields for code, status, and message. Several sample entries are shown.

```
2019-08-27 01:34:36 207,Success,Client 05a26a97-272a-4bc9-8f64-269d154b0e39 connected
2019-08-27 01:33:33 208,Warning,Client ec53d95c-1c88-41ae-8174-92104212de5d disconnected
2019-08-27 01:35:44 209,Success,Transaction 10d65890-b003-48f8-9cf8-9c74b51189c8 succeeded
2019-08-27 01:38:22 302,Error,Application could not connect to database
2019-08-27 01:31:34 303,Error,Application lost connection to database
```

### Upload and parse a sample log

We provide one of the log files and can see the events that it will be collecting. In this case, **New line** is a sufficient delimiter. If a single entry in the log could span multiple lines though, a timestamp delimiter would need to be used.

2. Select record delimiter

New line  
 Timestamp

**Record# 1**  
2019-08-27 01:34:36 207,Success,Client 05a26a97-272a-4bc9-8f64-269d154b0e39 connected

**Record# 2**  
2019-08-27 01:33:33 208,Warning,Client ec53d95c-1c88-41ae-8174-92104212de5d disconnected

**Record# 3**  
2019-08-27 01:35:44 209,Success,Transaction 10d65890-b003-48f8-9fcf-9c74b51189c8 succeeded

**Record# 4**  
2019-08-27 01:38:22 302,Error,Application could not connect to database

**Record# 5**  
2019-08-27 01:31:34 303,Error,Application lost connection to database

### Add log collection paths

The log files will be located in *C:\MyApp\Logs*. A new file will be created each day with a name that includes the date in the pattern *appYYYYMMDD.log*. A sufficient pattern for this log would be *C:\MyApp\Logs\\*.log*.

3. Add log collection paths

Windows  Linux

C:\MyApp\Logs\app\*.log +

C:\MyApp\Logs\app\*.log Windows Remove

### Provide a name and description for the log

We use a name of *MyApp\_CL* and type in a Description.

4. Add name and description

Name:

Description

### Validate that the custom logs are being collected

We use a simple query of *MyApp\_CL* to return all records from the collected log.

MyApp\_CL

Completed. Showing results from the last 7 days.

00:00:00.920 17 records Display time (UTC-07:00) ▾

Drag a column header and drop it here to group by that column

TimeGenerated [Local T... Computer RawData Type \_ResourceId

TimeGenerated [Local T...	Computer	RawData	Type	_ResourceId
2019-08-28T11:22:01.000	srv01.contoso.com	2019-08-27 01:34:36 207,Success,Client 05a26a97-272a-4bc9-8f64-26...	MyApp_CL	/subscriptions/4e56605e-4b16-4baa-9358-d...
2019-08-28T11:22:01.000	srv01.contoso.com	2019-08-27 01:33:33 208,Warning,Client ec53d95c-1c88-41ae-8174-92...	MyApp_CL	/subscriptions/4e56605e-4b16-4baa-9358-d...
2019-08-28T11:22:01.000	srv01.contoso.com	2019-08-27 01:35:44 209,Success,Transaction 10d65890-b003-48f8-9c...	MyApp_CL	/subscriptions/4e56605e-4b16-4baa-9358-d...
...				
TenantId	73308ade-39f9-41c5-aa82-28c9b614d0e2			
SourceSystem	OpsManager			
MG	2aeebe1c-d431-28e2-bb1a-603e93db2b48			
ManagementGroupName	AOI-73308ade-39f9-41c5-aa82-28c9b614d0e2			
TimeGenerated [UTC]	2019-08-28T18:22:01Z			
Computer				
RawData	2019-08-27 01:35:44 209,Success,Transaction 10d65890-b003-48f8-9fcf-9c74b51189c8 succeeded			
Type	MyApp_CL			
_ResourceId	/subscriptions/4e56605e-4b16-4baa-9358-dbxxxxxxxx/resourcegroups/myresourcegroup/providers/microsoft.compute/virtualmachines/srv01			

## Alternatives to custom logs

While custom logs are useful if your data fits the criteria listed, there are cases where you need another strategy:

- The data doesn't fit the required structure, such as having the timestamp in a different format.
- The log file doesn't adhere to requirements such as file encoding or an unsupported folder structure.
- The data requires preprocessing or filtering before collection.

In the cases where your data can't be collected with custom logs, consider the following alternate strategies:

- Use a custom script or other method to write data to [Windows Events](#) or [Syslog](#), which are collected by Azure Monitor.
- Send the data directly to Azure Monitor by using [HTTP Data Collector API](#).

## Next steps

- See [Parse text data in Azure Monitor](#) for methods to parse each imported log entry into multiple properties.
- Learn about [log queries](#) to analyze the data collected from data sources and solutions.

# Create custom fields in a Log Analytics workspace in Azure Monitor (Preview)

9/21/2022 • 7 minutes to read • [Edit Online](#)

## NOTE

This article describes how to parse text data in a Log Analytics workspace as it's collected. We recommend parsing text data in a query filter after it's collected following the guidance described in [Parse text data in Azure Monitor](#). It provides several advantages over using custom fields.

## IMPORTANT

Custom fields increases the amount of data collected in the Log Analytics workspace which can increase your cost. See [Azure Monitor Logs pricing details](#) for details.

The **Custom Fields** feature of Azure Monitor allows you to extend existing records in your Log Analytics workspace by adding your own searchable fields. Custom fields are automatically populated from data extracted from other properties in the same record.



For example, the sample record below has useful data buried in the event description. Extracting this data into a separate property makes it available for such actions as sorting and filtering.

TenantId	SourceSystem	TimeGenerated [Local Time]	Source	EventLog	Compute
<GUID Removed>	OpsManager	2019-03-29T13:28:27.397	Service Control Manager	System	
...	TenantId	<GUID Removed>			
	SourceSystem	OpsManager			
	TimeGenerated [UTC]	2019-03-29T20:28:27.397Z			
	Source	Service Control Manager			
	EventLog	System			
	Computer	contoso-srv-01.contoso.com			
	EventLevel	4			
	EventLevelName	Information			
	ParameterXml	<Param>WMI Performance Adapter</Param><Param>running</Param><Param>-</Param>			
	EventData	<DataItem type="System.XmlData" time="2019-03-29T20:28:27.3969411+00:00" sourceHealthServiceId="0917C52B-0E75			
	EventID	7,036			
	RenderedDescription	The WMI Performance Adapter service entered the running state.			
	EventCategory	0			
	UserName	N/A			
	MG	00000000-0000-0000-000000000001			
	ManagementGroupName	<GUID Removed>			
	Service_CF	WMI Performance Adapter			
	Type	Event			
	_ResourceId	<GUID Removed>			

#### NOTE

In the Preview, you are limited to 500 custom fields in your workspace. This limit will be expanded when this feature reaches general availability.

## Creating a custom field

When you create a custom field, Log Analytics must understand which data to use to populate its value. It uses a technology from Microsoft Research called FlashExtract to quickly identify this data. Rather than requiring you to provide explicit instructions, Azure Monitor learns about the data you want to extract from examples that you provide.

The following sections provide the procedure for creating a custom field. At the bottom of this article is a walkthrough of a sample extraction.

#### NOTE

The custom field is populated as records matching the specified criteria are added to the Log Analytics workspace, so it will only appear on records collected after the custom field is created. The custom field will not be added to records that are already in the data store when it's created.

### Step 1 – Identify records that will have the custom field

The first step is to identify the records that will get the custom field. You start with a [standard log query](#) and then select a record to act as the model that Azure Monitor will learn from. When you specify that you are going to extract data into a custom field, the **Field Extraction Wizard** is opened where you validate and refine the criteria.

1. Go to **Logs** and use a [query to retrieve the records](#) that will have the custom field.
2. Select a record that Log Analytics will use to act as a model for extracting data to populate the custom field.  
You will identify the data that you want to extract from this record, and Log Analytics will use this information to determine the logic to populate the custom field for all similar records.

3. Right-click on the record, and select **Extract fields from**.
4. The **Field Extraction Wizard** is opened, and the record you selected is displayed in the **Main Example** column. The custom field will be defined for those records with the same values in the properties that are selected.
5. If the selection is not exactly what you want, select additional fields to narrow the criteria. In order to change the field values for the criteria, you must cancel and select a different record matching the criteria you want.

### **Step 2 - Perform initial extract.**

Once you've identified the records that will have the custom field, you identify the data that you want to extract. Log Analytics will use this information to identify similar patterns in similar records. In the step after this you will be able to validate the results and provide further details for Log Analytics to use in its analysis.

1. Highlight the text in the sample record that you want to populate the custom field. You will then be presented with a dialog box to provide a name and data type for the field and to perform the initial extract. The characters \_CF will automatically be appended.
2. Click **Extract** to perform an analysis of collected records.
3. The **Summary** and **Search Results** sections display the results of the extract so you can inspect its accuracy. **Summary** displays the criteria used to identify records and a count for each of the data values identified. **Search Results** provides a detailed list of records matching the criteria.

### **Step 3 – Verify accuracy of the extract and create custom field**

Once you have performed the initial extract, Log Analytics will display its results based on data that has already been collected. If the results look accurate then you can create the custom field with no further work. If not, then you can refine the results so that Log Analytics can improve its logic.

1. If any values in the initial extract aren't correct, then click the **Edit** icon next to an inaccurate record and select **Modify this highlight** in order to modify the selection.
2. The entry is copied to the **Additional examples** section underneath the **Main Example**. You can adjust the highlight here to help Log Analytics understand the selection it should have made.
3. Click **Extract** to use this new information to evaluate all the existing records. The results may be modified for records other than the one you just modified based on this new intelligence.
4. Continue to add corrections until all records in the extract correctly identify the data to populate the new custom field.
5. Click **Save Extract** when you are satisfied with the results. The custom field is now defined, but it won't be added to any records yet.
6. Wait for new records matching the specified criteria to be collected and then run the log search again. New records should have the custom field.
7. Use the custom field like any other record property. You can use it to aggregate and group data and even use it to produce new insights.

## **Viewing custom fields**

You can view a list of all custom fields in your management group from the **Advanced Settings** menu of your Log Analytics workspace in the Azure portal. Select **Data** and then **Custom fields** for a list of all custom fields in your workspace.

The screenshot shows the 'Advanced settings' interface. On the left, there's a sidebar with 'Connected Sources', 'Data', 'Computer Groups', 'Custom Fields' (which is selected and highlighted in blue), 'Custom Logs', and 'Syslog'. On the right, under 'Manage custom fields (1 fields used)', there's a table with one row:

FIELD NAME	LOG TYPE	FIELD TYPE
Service_CF	Event	Text <a href="#">Go to Remove</a>

## Removing a custom field

There are two ways to remove a custom field. The first is the **Remove** option for each field when viewing the complete list as described above. The other method is to retrieve a record and click the button to the left of the field. The menu will have an option to remove the custom field.

## Sample walkthrough

The following section walks through a complete example of creating a custom field. This example extracts the service name in Windows events that indicate a service changing state. This relies on events created by Service Control Manager during system startup on Windows computers. If you want to follow this example, you must be [collecting Information events for the System log](#).

We enter the following query to return all events from Service Control Manager that have an Event ID of 7036 which is the event that indicates a service starting or stopping.

The screenshot shows the Kibana interface. At the top, there are buttons for 'Run' (highlighted in blue), 'Time range: Last 24 hours', 'Save', 'Copy link', 'Export', 'New alert rule', and 'Pin'. The search bar contains the query:

```
Event
| where Source == "Service Control Manager"
| where EventID == 7036
```

Below the search bar, it says 'Completed. Showing results from the last 24 hours.' and shows '00:00:00.957' and '54 records'. It also shows 'Display time (UTC-07:00)'. The results table has columns: TenantId, SourceSystem, TimeGenerated [Local Time], Source, and EventLog. The data shows three rows of results:

TenantId	SourceSystem	TimeGenerated [Local Time]	Source	EventLog
<GUID Removed>	OpsManager	2019-03-29T10:12:54.803	Service Control Manager	System
<GUID Removed>	OpsManager	2019-03-29T10:16:33.150	Service Control Manager	System
<GUID Removed>	OpsManager	2019-03-29T10:16:36.060	Service Control Manager	System

We then right-click on any record with event ID 7036 and select **Extract fields from 'Event'**.

Completed. Showing results from the last 24 hours.

00:00:00.957 54 records

Display time (UTC-07:00)

TenantId	SourceSystem	TimeGenerated [Local Time]	Source	EventLog
<GUID Removed>	OpsManager	2019-03-29T10:12:54.803	Service Control Manager	System
<GUID Removed>	Manager	2019-03-29T10:16:33.150	Service Control Manager	System
<GUID Removed>	Manager	2019-03-29T10:16:36.060	Service Control Manager	System

The **Field Extraction Wizard** opens with the **EventLog** and **EventID** fields selected in the **Main Example** column. This indicates that the custom field will be defined for events from the System log with an event ID of 7036. This is sufficient so we don't need to select any other fields.

### MAIN EXAMPLE

Event		
<b>FILTER</b>	<b>FIELD NAME</b>	<b>VALUE</b>
<input type="checkbox"/>	SourceSystem	: OpsManager
<input type="checkbox"/>	TimeGenerated	: 2019-03-29T17:40:07.15Z
<input type="checkbox"/>	Source	: Service Control Manager
<input checked="" type="checkbox"/>	EventLog	: System
<input type="checkbox"/>	Computer	: contoso-srv-01.contoso.com
<input type="checkbox"/>	EventLevel	: 4
<input type="checkbox"/>	EventLevelName	: Information
<input type="checkbox"/>	ParameterXml	: <Param>Software Protection</Param><Param>stopped</Param><Param>-</Param><DataItem type="System.XmlData" time="2019-03-29T17:40:07.1493210+00:00" sourceHealthServiceId="0917C52B-0E75-D88B-6452-771406B8EFD2"><EventData xmlns="http://schemas.microsoft.com/oft.com/win/2004/08/events/event"><Data Name="param1">Software Protection</Data><Data Name="param2">stopped</Data><Binary>7300700070007300760063002F0031000000</Binary></EventData></DataItem>
<input type="checkbox"/>	EventData	:
<input checked="" type="checkbox"/>	EventID	: 7036
<input type="checkbox"/>	RenderedDescription	: The Software Protection service entered the stopped state.
<input type="checkbox"/>	AzureDeploymentID	:
<input type="checkbox"/>	Role	:
<input type="checkbox"/>	EventCategory	: 0
<input type="checkbox"/>	UserName	: N/A
<input type="checkbox"/>	Message	:
<input type="checkbox"/>	ManagementGroupName	: <GUID Removed>
<input type="checkbox"/>	DataItem_CF	:

We highlight the name of the service in the **RenderedDescription** property and use **Service** to identify the service name. The custom field will be called **Service\_CF**. The field type in this case is a string, so we can leave that unchanged.

<input checked="" type="checkbox"/> EventID	:	7036
<input type="checkbox"/> RenderedDescription	:	The Software Protection service entered the stopped state.
<input type="checkbox"/> AzureDeploymentID	:	Field value : Software Protection
<input type="checkbox"/> Role	:	Field Title : <input type="text" value="Service_CF"/>
<input type="checkbox"/> EventCategory	:	Field Type : <input type="text" value="Text"/>
<input type="checkbox"/> UserName	:	<input type="button" value="Close"/>
<input type="checkbox"/> Message	:	<input type="button" value="Extract"/>
<input type="checkbox"/> ManagementGroupName	:	<GU>
<input type="checkbox"/> DataItem_CF	:	

We see that the service name is identified properly for some records but not for others. The **Search Results** show that part of the name for the **WMI Performance Adapter** wasn't selected. The **Summary** shows that one record identified **Modules Installer** instead of **Windows Modules Installer**.

SEARCH RESULTS	SUMMARY
<b>Modules Installer</b> The Windows Modules Installer service entered the stopped state.	[ <input type="checkbox"/> hide tips] Condition Event   where EventLog == "System"   where EventID == 7036   limit 100 Service_CF (4 values) <input type="text" value="Modules Installer (1 matches)"/> <input type="text" value="Performance Adapter (39 matches)"/> <input type="text" value="Windows Update (12 matches)"/> <input type="text" value="Software Protection (0 matches)"/>
<b>Performance Adapter</b> The WMI Performance Adapter service entered the running state.	
<b>Performance Adapter</b> The WMI Performance Adapter service entered the stopped state.	
<b>Performance Adapter</b> The WMI Performance Adapter service entered the running state.	
<b>Performance Adapter</b> The WMI Performance Adapter service entered the stopped state.	
<b>Windows Update</b> The Windows Update service entered the stopped state.	

We start with the **WMI Performance Adapter** record. We click its edit icon and then **Modify this highlight**.

<b>Performance Adapter</b> The WMI Performance Adapter service entered the running state.	... <input type="button" value="Modify this highlight..."/>
<b>Performance Adapter</b> The WMI Performance Adapter service entered the stopped state.	<input type="button" value="Ignore results like these."/>  <input type="text" value="Modules Installer (1 matches)"/> <input type="text" value="Performance Adapter (39 matches)"/> <input type="text" value="Windows Update (12 matches)"/> <input type="text" value="Software Protection (0 matches)"/>

We increase the highlight to include the word **WMI** and then rerun the extract.

Additional examples
New - Highlight text to mark a custom field. Click on existing highlights to remove.
<input checked="" type="checkbox"/> RenderedDescription : The WMI Performance Adapter service entered the running state.

We can see that the entries for **WMI Performance Adapter** have been corrected, and Log Analytics also used that information to correct the records for **Windows Module Installer**.

SEARCH RESULTS		SUMMARY
Windows Modules Installer	(1)	[x] hide tips
The Windows Modules Installer service entered the stopped state.		Condition
WMI Performance Adapter	(1)	Event   where EventLog == "System"   where EventID == 7036   limit 100
The WMI Performance Adapter service entered the running state.		Service_CF (4 values)
WMI Performance Adapter	(1)	Windows Modules Installer (1 matches)
The WMI Performance Adapter service entered the stopped state.		WMI Performance Adapter (40 matches)
WMI Performance Adapter	(1)	Windows Update (13 matches)
The WMI Performance Adapter service entered the running state.		Software Protection (5 matches)
WMI Performance Adapter	(1)	
The WMI Performance Adapter service entered the stopped state.		

We can now run a query that verifies Service\_CF is created but is not yet added to any records. That's because the custom field doesn't work against existing records so we need to wait for new records to be collected.

The screenshot shows the Kibana interface with a search results table. The table has two columns: 'Service\_CF' and 'count\_'. There is one row with the value '1,725' under 'count\_'. The table header includes column sorting arrows. At the top, there is a 'Run' button, a time range selector set to 'Last 24 hours', and various action buttons like 'Save', 'Copy link', 'Export', 'New alert rule', and 'Pin'. The status bar at the bottom right shows '00:00:00.770' and '1 records'.

Service_CF	count_
>	1,725

After some time has passed so new events are collected, we can see that the Service\_CF field is now being added to records that match our criteria.

The screenshot shows the Kibana interface with a search results table. The table has two columns: 'Service\_CF' and 'count\_'. There are three rows: one with '1,730' under 'count\_' and two others with '2' under 'count\_'. The table header includes column sorting arrows. At the top, there is a 'Run' button, a time range selector set to 'Last 24 hours', and various action buttons like 'Save', 'Copy link', 'Export', 'New alert rule', and 'Pin'. The status bar at the bottom right shows '00:00:09.985' and '3 records'.

Service_CF	count_
>	1,730
> WMI Performance Adapter	2
> Print Spooler	2

We can now use the custom field like any other record property. To illustrate this, we create a query that groups by the new Service\_CF field to inspect which services are the most active.

TenantId	SourceSystem	TimeGenerated [Local Time]	Source	EventLog	Compute
<GUID Removed>	OpsManager	2019-03-29T13:28:27.397	Service Control Manager	System	
...					
TenantId	<GUID Removed>				
SourceSystem	OpsManager				
TimeGenerated [UTC]	2019-03-29T20:28:27.397Z				
Source	Service Control Manager				
EventLog	System				
Computer	contoso-srv-01.contoso.com				
EventLevel	4				
EventLevelName	Information				
ParameterXml	<Param>WMI Performance Adapter</Param><Param>running</Param><Param>-</Param>				
EventData	<DataItem type="System.XmlData" time="2019-03-29T20:28:27.3969411+00:00" sourceHealthServiceId="0917C52B-0E75				
EventID	7,036				
RenderedDescription	The WMI Performance Adapter service entered the running state.				
EventCategory	0				
UserName	N/A				
MG	00000000-0000-0000-0000-000000000001				
ManagementGroupName	<GUID Removed>				
Service_CF	WMI Performance Adapter				
Type	Event				
_Resourceid	<GUID Removed>				

## Next steps

- Learn about [log queries](#) to build queries using custom fields for criteria.
- Monitor [custom log files](#) that you parse using custom fields.

# Troubleshooting the Log Analytics VM extension in Azure Monitor

9/21/2022 • 2 minutes to read • [Edit Online](#)

This article provides help troubleshooting errors you might experience with the Log Analytics VM extension for Windows and Linux virtual machines running on Microsoft Azure, and suggests possible solutions to resolve them.

To verify the status of the extension, perform the following steps from the Azure portal.

1. Sign into the [Azure portal](#).
2. In the Azure portal, click **All services**. In the list of resources, type **virtual machines**. As you begin typing, the list filters based on your input. Select **Virtual machines**.
3. In your list of virtual machines, find and select it.
4. On the virtual machine, click **Extensions**.
5. From the list, check to see if the Log Analytics extension is enabled or not. For Linux, the agent is listed as **OMSAgentforLinux** and for Windows, the agent is listed as **MicrosoftMonitoringAgent**.

The screenshot shows the Azure portal interface for a virtual machine named "DC01 - Extensions". The left sidebar contains navigation links: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, SETTINGS, Networking, Disks, Size, and Extensions. The "Extensions" link is highlighted with a blue selection bar. The main content area is titled "+ Add" and shows a table of extensions. The table has columns: NAME, TYPE, VER..., and STATUS. It lists two entries:

NAME	TYPE	VER...	STATUS
IaaS Diagnostics	Microsoft.Azure.Diagnostics.IaaSDi...	1.*	Transitioning
Microsoft Monitoring Agent	Microsoft.EnterpriseCloud.Monitori...	1.*	Provisioning succeeded

6. Click on the extension to view details.

MicrosoftMonitoringAgent	
DC01	
 Uninstall	
TYPE	Microsoft.EnterpriseCloud.Monitoring.MicrosoftMoni...
VERSION	1.0.11081.2
STATUS	Provisioning succeeded
STATUS LEVEL	Info
STATUS MESSAGE	Latest configuration has been applied to the Microsof...
HANDLER STATUS	Ready
HANDLER STATUS LEVEL	Info

## Troubleshooting Azure Windows VM extension

If the *Microsoft Monitoring Agent* VM extension is not installing or reporting, you can perform the following steps to troubleshoot the issue.

1. Check if the Azure VM agent is installed and working correctly by using the steps in [KB 2965986](#).
  - You can also review the VM agent log file `C:\WindowsAzure\logs\WaAppAgent.log`
  - If the log does not exist, the VM agent is not installed.
  - [Install the Azure VM Agent](#)
2. Review the Microsoft Monitoring Agent VM extension log files in  
`C:\Packages\Plugins\Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent`
3. Ensure the virtual machine can run PowerShell scripts
4. Ensure permissions on `C:\Windows\temp` haven't been changed
5. View the status of the Microsoft Monitoring Agent by typing the following in an elevated PowerShell window on the virtual machine  
`(New-Object -ComObject 'AgentConfigManager.MgmtSvcCfg').GetCloudWorkspaces() | Format-List`
6. Review the Microsoft Monitoring Agent setup log files in  
`C:\WindowsAzure\Logs\Plugins\Microsoft.EnterpriseCloud.Monitoring.MicrosoftMonitoringAgent\1.0.18053.0\`.  
 Note that this path will change based on the version number of the agent.

For more information, see [troubleshooting Windows extensions](#).

## Troubleshooting Linux VM extension

### NOTE

As part of the ongoing transition from Microsoft Operations Management Suite to Azure Monitor, the Operations Management Suite Agent for Windows or Linux will be referred to as the Log Analytics agent for Windows and Log Analytics agent for Linux.

If the *Log Analytics agent for Linux* VM extension is not installing or reporting, you can perform the following steps to troubleshoot the issue.

1. If the extension status is *Unknown* check if the Azure VM agent is installed and working correctly by reviewing the VM agent log file `/var/log/waagent.log`
  - If the log does not exist, the VM agent is not installed.
  - [Install the Azure VM Agent on Linux VMs](#)
2. For other unhealthy statuses, review the Log Analytics agent for Linux VM extension logs files in `/var/log/azure/Microsoft.EnterpriseCloud.Monitoring.OmsAgentForLinux/*/extension.log` and `/var/log/azure/Microsoft.EnterpriseCloud.Monitoring.OmsAgentForLinux/*/CommandExecution.log`
3. If the extension status is healthy, but data is not being uploaded review the Log Analytics agent for Linux log files in `/var/opt/microsoft/omsagent/log/omsagent.log`

## Next steps

For additional troubleshooting guidance related to the Log Analytics agent for Linux, see [Troubleshoot Azure Log Analytics Linux Agent](#).

# Troubleshoot issues with the Log Analytics agent for Linux

9/21/2022 • 18 minutes to read • [Edit Online](#)

This article provides help in troubleshooting errors you might experience with the Log Analytics agent for Linux in Azure Monitor and suggests possible solutions to resolve them.

## Log Analytics Troubleshooting Tool

The Log Analytics agent for Linux Troubleshooting Tool is a script designed to help find and diagnose issues with the Log Analytics agent. It's automatically included with the agent upon installation. Running the tool should be the first step in diagnosing an issue.

### Use the Troubleshooting Tool

To run the Troubleshooting Tool, paste the following command into a terminal window on a machine with the Log Analytics agent:

```
sudo /opt/microsoft/omsagent/bin/troubleshooter
```

### Manual installation

The Troubleshooting Tool is automatically included when the Log Analytics agent is installed. If installation fails in any way, you can also install the tool manually:

1. Ensure that the [GNU Project Debugger \(GDB\)](#) is installed on the machine because the troubleshooter relies on it.
2. Copy the troubleshooter bundle onto your machine:

```
wget https://raw.github.com/microsoft/OMS-Agent-for-Linux/master/source/code/troubleshooter/omsagent_tst.tar.gz
```

3. Unpack the bundle: `tar -xvf omsagent_tst.tar.gz`
4. Run the manual installation: `sudo ./install_tst`

### Scenarios covered

The Troubleshooting Tool checks the following scenarios:

- The agent is unhealthy; the heartbeat doesn't work properly.
- The agent doesn't start or can't connect to Log Analytics.
- The agent Syslog isn't working.
- The agent has high CPU or memory usage.
- The agent has installation issues.
- The agent custom logs aren't working.
- Agent logs can't be collected.

For more information, see the [Troubleshooting Tool documentation on GitHub](#).

#### NOTE

Run the Log Collector tool when you experience an issue. Having the logs initially will help our support team troubleshoot your issue faster.

## Purge and reinstall the Linux agent

A clean reinstall of the agent fixes most issues. This task might be the first suggestion from our support team to get the agent into an uncorrupted state. Running the Troubleshooting Tool and Log Collector tool and attempting a clean reinstall helps to solve issues more quickly.

1. Download the purge script:

```
$ wget https://raw.githubusercontent.com/microsoft/OMS-Agent-for-Linux/master/tools/purge_omsagent.sh
```

2. Run the purge script (with sudo permissions):

```
$ sudo sh purge_omsagent.sh
```

## Important log locations and the Log Collector tool

FILE	PATH
Log Analytics agent for Linux log file	/var/opt/microsoft/omsagent/<workspace id>/log/omsagent.log
Log Analytics agent configuration log file	/var/opt/microsoft/omsconfig/omsconfig.log

We recommend that you use the Log Collector tool to retrieve important logs for troubleshooting or before you submit a GitHub issue. For more information about the tool and how to run it, see [OMS Linux Agent Log Collector](#).

## Important configuration files

CATEGORY	FILE LOCATION
Syslog	/etc/syslog-ng/syslog-ng.conf or /etc/rsyslog.conf or /etc/rsyslog.d/95-omsagent.conf
Performance, Nagios, Zabbix, Log Analytics output and general agent	/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf
Extra configurations	/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.d/*.conf

### NOTE

Editing configuration files for performance counters and Syslog is overwritten if the collection is configured from the [agent's configuration](#) in the Azure portal for your workspace. To disable configuration for all agents, disable collection from [Agents configuration](#). For a single agent, run the following script:

```
sudo /opt/microsoft/omsconfig/Scripts/OMS_MetaConfigHelper.py --disable && sudo rm  
/etc/opt/omi/conf/omsconfig/configuration/Current.mof*  
/etc/opt/omi/conf/omsconfig/configuration/Pending.mof*
```

## Installation error codes

ERROR CODE	MEANING

ERROR CODE	MEANING
NOT_DEFINED	Because the necessary dependencies aren't installed, the auoms audited plug-in won't be installed. Installation of auoms failed. Install package audited.
2	Invalid option provided to the shell bundle. Run <code>sudo sh ./omsagent-*.universal*.sh --help</code> for usage.
3	No option provided to the shell bundle. Run <code>sudo sh ./omsagent-*.universal*.sh --help</code> for usage.
4	Invalid package type <i>or</i> invalid proxy settings. The omsagent-rpm.sh packages can only be installed on RPM-based systems. The omsagent-deb.sh packages can only be installed on Debian-based systems. We recommend that you use the universal installer from the <a href="#">latest release</a> . Also review to verify your proxy settings.
5	The shell bundle must be executed as root <i>or</i> there was a 403 error returned during onboarding. Run your command by using <code>sudo</code> .
6	Invalid package architecture <i>or</i> there was a 200 error returned during onboarding. The omsagent-*x64.sh packages can only be installed on 64-bit systems. The omsagent-*x86.sh packages can only be installed on 32-bit systems. Download the correct package for your architecture from the <a href="#">latest release</a> .
17	Installation of OMS package failed. Look through the command output for the root failure.
18	Installation of OMSConfig package failed. Look through the command output for the root failure.
19	Installation of OMI package failed. Look through the command output for the root failure.
20	Installation of SCX package failed. Look through the command output for the root failure.
21	Installation of Provider kits failed. Look through the command output for the root failure.
22	Installation of bundled package failed. Look through the command output for the root failure
23	SCX or OMI package already installed. Use <code>--upgrade</code> instead of <code>--install</code> to install the shell bundle.
30	Internal bundle error. File a <a href="#">GitHub issue</a> with details from the output.
55	Unsupported openssl version <i>or</i> can't connect to Azure Monitor <i>or</i> dpkg is locked <i>or</i> missing curl program.

ERROR CODE	MEANING
61	Missing Python ctypes library. Install the Python ctypes library or package (python-ctypes).
62	Missing tar program. Install tar.
63	Missing sed program. Install sed.
64	Missing curl program. Install curl.
65	Missing gpg program. Install gpg.

## Onboarding error codes

ERROR CODE	MEANING
2	Invalid option provided to the omsadmin script. Run <pre>sudo sh /opt/microsoft/omsagent/bin/omsadmin.sh -h</pre> for usage.
3	Invalid configuration provided to the omsadmin script. Run <pre>sudo sh /opt/microsoft/omsagent/bin/omsadmin.sh -h</pre> for usage.
4	Invalid proxy provided to the omsadmin script. Verify the proxy and see our <a href="#">documentation for using an HTTP proxy</a> .
5	403 HTTP error received from Azure Monitor. See the full output of the omsadmin script for details.
6	Non-200 HTTP error received from Azure Monitor. See the full output of the omsadmin script for details.
7	Unable to connect to Azure Monitor. See the full output of the omsadmin script for details.
8	Error onboarding to Log Analytics workspace. See the full output of the omsadmin script for details.
30	Internal script error. File a <a href="#">GitHub issue</a> with details from the output.
31	Error generating agent ID. File a <a href="#">GitHub issue</a> with details from the output.
32	Error generating certificates. See the full output of the omsadmin script for details.
33	Error generating metaconfiguration for omsconfig. File a <a href="#">GitHub issue</a> with details from the output.

ERROR CODE	MEANING
34	<p>Metaconfiguration generation script not present. Retry onboarding with</p> <pre>sudo sh /opt/microsoft/omsagent/bin/omsadmin.sh -w &lt;Workspace ID&gt; -s &lt;Workspace Key&gt;</pre>

## Enable debug logging

### OMS output plug-in debug

FluentD allows for plug-in-specific logging levels that allow you to specify different log levels for inputs and outputs. To specify a different log level for OMS output, edit the general agent configuration at

```
/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf
```

In the OMS output plug-in, before the end of the configuration file, change the `log_level` property from `info` to `debug`:

```
<match oms.** docker.**>
  type out_oms
  log_level debug
  num_threads 5
  buffer_chunk_limit 5m
  buffer_type file
  buffer_path /var/opt/microsoft/omsagent/<workspace id>/state/out_oms*.buffer
  buffer_queue_limit 10
  flush_interval 20s
  retry_limit 10
  retry_wait 30s
</match>
```

Debug logging allows you to see batched uploads to Azure Monitor separated by type, number of data items, and time taken to send.

Here's an example debug-enabled log:

```
Success sending oms.nagios x 1 in 0.14s
Success sending oms.omi x 4 in 0.52s
Success sending oms.syslog.authpriv.info x 1 in 0.91s
```

### Verbose output

Instead of using the OMS output plug-in, you can output data items directly to `stdout`, which is visible in the Log Analytics agent for Linux log file.

In the Log Analytics general agent configuration file at

```
/etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf
```

comment out the OMS output plug-in by adding a `#` in front of each line:

```
#<match oms.** docker.**>
#  type out_oms
#  log_level info
#  num_threads 5
#  buffer_chunk_limit 5m
#  buffer_type file
#  buffer_path /var/opt/microsoft/omsagent/<workspace id>/state/out_oms*.buffer
#  buffer_queue_limit 10
#  flush_interval 20s
#  retry_limit 10
#  retry_wait 30s
#</match>
```

Below the output plug-in, uncomment the following section by removing the `#` in front of each line:

```
<match **>
  type stdout
</match>
```

## Issue: Unable to connect through proxy to Azure Monitor

### Probable causes

- The proxy specified during onboarding was incorrect.
- The Azure Monitor and Azure Automation service endpoints aren't included in the approved list in your datacenter.

### Resolution

1. Reonboard to Azure Monitor with the Log Analytics agent for Linux by using the following command with the option `-v` enabled. It allows verbose output of the agent connecting through the proxy to Azure Monitor:  
`/opt/microsoft/omsagent/bin/omsadmin.sh -w <Workspace ID> -s <Workspace Key> -p <Proxy Conf> -v`
2. Review the section [Update proxy settings](#) to verify you've properly configured the agent to communicate through a proxy server.
3. Double-check that the endpoints outlined in the Azure Monitor [network firewall requirements](#) list are added to an allow list correctly. If you use Azure Automation, the necessary network configuration steps are also linked above.

## Issue: You receive a 403 error when trying to onboard

### Probable causes

- Date and time are incorrect on the Linux server.
- The workspace ID and workspace key aren't correct.

### Resolution

1. Check the time on your Linux server with the command `date`. If the time is +/- 15 minutes from the current time, onboarding fails. To correct this situation, update the date and/or time zone of your Linux server.
2. Verify that you've installed the latest version of the Log Analytics agent for Linux. The newest version now notifies you if time skew is causing the onboarding failure.
3. Reonboard by using the correct workspace ID and workspace key in the installation instructions earlier in this article.

## Issue: You see a 500 and 404 error in the log file right after onboarding

This is a known issue that occurs on the first upload of Linux data into a Log Analytics workspace. This issue doesn't affect data being sent or service experience.

## Issue: You see omiagent using 100% CPU

### Probable causes

A regression in nss-pem package [v1.0.3-5.el7](#) caused a severe performance issue. We've been seeing this issue come up a lot in Redhat/Centos 7.x distributions. To learn more about this issue, see [1667121 Performance regression in libcurl](#).

Performance-related bugs don't happen all the time, and they're difficult to reproduce. If you experience such an issue with omiagent, use the script `omiHighCPUDiagnostics.sh`, which will collect the stack trace of the omiagent when it exceeds a certain threshold.

1. Download the script:

```
wget https://raw.githubusercontent.com/microsoft/OMS-Agent-for-Linux/master/tools/LogCollector/source/omiHighCPUDiagnostics.sh
```

2. Run diagnostics for 24 hours with 30% CPU threshold:

```
bash omiHighCPUDiagnostics.sh --runtime-in-min 1440 --cpu-threshold 30
```

3. Callstack will be dumped in the `omiagent_trace` file. If you notice many curl and NSS function calls, follow these resolution steps.

### Resolution

1. Upgrade the nss-pem package to [v1.0.3-5.el7\\_6.1](#):

```
sudo yum upgrade nss-pem
```

2. If nss-pem isn't available for upgrade, which mostly happens on Centos, downgrade curl to 7.29.0-46. If you run "yum update" by mistake, curl will be upgraded to 7.29.0-51 and the issue will happen again:

```
sudo yum downgrade curl libcurl
```

3. Restart OMI:

```
sudo scxadmin -restart
```

## Issue: You're not seeing forwarded Syslog messages

### Probable causes

- The configuration applied to the Linux server doesn't allow collection of the sent facilities or log levels.
- Syslog isn't being forwarded correctly to the Linux server.
- The number of messages being forwarded per second is too great for the base configuration of the Log Analytics agent for Linux to handle.

### Resolution

- Verify the configuration in the Log Analytics workspace for Syslog has all the facilities and the correct log levels. Review [configure Syslog collection in the Azure portal](#).
- Verify the native Syslog messaging daemons (`rsyslog`, `syslog-ng`) can receive the forwarded messages.
- Check firewall settings on the Syslog server to ensure that messages aren't being blocked.
- Simulate a Syslog message to Log Analytics by using a `logger` command:  

```
logger -p local0.err "This is my test message"
```

# Issue: You're receiving Errno address already in use in omsagent log file

You see

```
[error]: unexpected error error_class=Errno::EADDRINUSE error=#<Errno::EADDRINUSE: Address already in use - bind(2) for "127.0.0.1" port 25224>
```

in omsagent.log.

## Probable causes

This error indicates that the Linux diagnostic extension (LAD) is installed side by side with the Log Analytics Linux VM extension. It's using the same port for Syslog data collection as omsagent.

## Resolution

1. As root, execute the following commands. Note that 25224 is an example, and it's possible that in your environment you see a different port number used by LAD.

```
/opt/microsoft/omsagent/bin/configure_syslog.sh configure LAD 25229  
sed -i -e 's/25224/25229/' /etc/opt/microsoft/omsagent/LAD/conf/omsagent.d/syslog.conf
```

You then need to edit the correct `rsyslogd` or `syslog_ng` config file and change the LAD-related configuration to write to port 25229.

2. If the VM is running `rsyslogd`, the file to be modified is `/etc/rsyslog.d/95-omsagent.conf` (if it exists, else `/etc/rsyslog`). If the VM is running `syslog_ng`, the file to be modified is `/etc/syslog-ng/syslog-ng.conf`.
3. Restart omsagent `sudo /opt/microsoft/omsagent/bin/service_control restart`.
4. Restart the Syslog service.

# Issue: You're unable to uninstall omsagent using the purge option

## Probable causes

- The Linux diagnostic extension is installed.
- The Linux diagnostic extension was installed and uninstalled, but you still see an error about omsagent being used by mdsd and it can't be removed.

## Resolution

1. Uninstall the Linux diagnostic extension.
2. Remove Linux diagnostic extension files from the machine if they're present in the following location:  
`/var/lib/waagent/Microsoft.Azure.Diagnostics.LinuxDiagnostic-<version>/` and  
`/var/opt/microsoft/omsagent/LAD/`.

# Issue: You can't see any Nagios data

## Probable causes

- The omsagent user doesn't have permissions to read from the Nagios log file.
- The Nagios source and filter haven't been uncommented from the omsagent.conf file.

## Resolution

1. Add the omsagent user to read from the Nagios file by following these [instructions](#).
2. In the Log Analytics agent for Linux general configuration file at  
`/etc/opt/microsoft/omsagent/<workspace_id>/conf/omsagent.conf`, ensure that *both* the Nagios source and

filter are uncommented.

```
<source>
  type tail
  path /var/log/nagios/nagios.log
  format none
  tag oms.nagios
</source>

<filter oms.nagios>
  type filter_nagios_log
</filter>
```

## Issue: You aren't seeing any Linux data

### Probable causes

- Onboarding to Azure Monitor failed.
- Connection to Azure Monitor is blocked.
- Virtual machine was rebooted.
- OMI package was manually upgraded to a newer version compared to what was installed by the Log Analytics agent for Linux package.
- OMI is frozen, blocking the OMS agent.
- DSC resource logs *class not found* error in `omsconfig.log` log file.
- Log Analytics agent for data is backed up.
- DSC logs *Current configuration does not exist. Execute Start-DscConfiguration command with -Path parameter to specify a configuration file and create a current configuration first.* in `omsconfig.log` log file, but no log message exists about `PerformRequiredConfigurationChecks` operations.

### Resolution

1. Install all dependencies like the `auditd` package.
2. Check if onboarding to Azure Monitor was successful by checking if the following file exists:  
`/etc/opt/microsoft/omsagent/<workspace_id>/conf/omsadmin.conf`. If it wasn't, reonboard by using the `omsadmin.sh` command-line [instructions](#).
3. If you're using a proxy, check the preceding proxy troubleshooting steps.
4. In some Azure distribution systems, the `omid` OMI server daemon doesn't start after the virtual machine is rebooted. If this is the case, you won't see Audit, ChangeTracking, or UpdateManagement solution-related data. The workaround is to manually start the OMI server by running  
`sudo /opt/omi/bin/service_control restart`.
5. After the OMI package is manually upgraded to a newer version, it must be manually restarted for the Log Analytics agent to continue functioning. This step is required for some distros where the OMI server doesn't automatically start after it's upgraded. Run `sudo /opt/omi/bin/service_control restart` to restart the OMI.

In some situations, the OMI can become frozen. The OMS agent might enter a blocked state waiting for the OMI, which blocks all data collection. The OMS agent process will be running but there will be no activity, which is evidenced by no new log lines (such as sent heartbeats) present in `omsagent.log`. Restart the OMI with `sudo /opt/omi/bin/service_control restart` to recover the agent.

6. If you see a DSC resource *class not found* error in `omsconfig.log`, run

```
sudo /opt/omi/bin/service_control restart
```

7. In some cases, when the Log Analytics agent for Linux can't talk to Azure Monitor, data on the agent is backed up to the full buffer size of 50 MB. The agent should be restarted by running the following command: `/opt/microsoft/omsagent/bin/service_control restart`.

**NOTE**

This issue is fixed in agent version 1.1.0-28 or later.

- If the `omsconfig.log` log file doesn't indicate that `PerformRequiredConfigurationChecks` operations are running periodically on the system, there might be a problem with the cron job/service. Make sure the cron job exists under `/etc/cron.d/OMSConsistencyInvoker`. If needed, run the following commands to create the cron job:

```
mkdir -p /etc/cron.d/
echo "**/15 * * * * omsagent /opt/omi/bin/OMSConsistencyInvoker >/dev/null 2>&1" | sudo tee
/etc/cron.d/OMSConsistencyInvoker
```

- Also, make sure the cron service is running. You can use `service cron status` with Debian, Ubuntu, and SUSE or `service crond status` with RHEL, CentOS, and Oracle Linux to check the status of this service. If the service doesn't exist, you can install the binaries and start the service by using the following instructions:

**Ubuntu/Debian**

```
# To Install the service binaries
sudo apt-get install -y cron
# To start the service
sudo service cron start
```

**SUSE**

```
# To Install the service binaries
sudo zypper in cron -y
# To start the service
sudo systemctl enable cron
sudo systemctl start cron
```

**RHEL/CentOS**

```
# To Install the service binaries
sudo yum install -y crond
# To start the service
sudo service crond start
```

**Oracle Linux**

```
# To Install the service binaries
sudo yum install -y crondie
# To start the service
sudo service crond start
```

**Issue:** When you configure collection from the portal for Syslog or

# Linux performance counters, the settings aren't applied

## Probable causes

- The Log Analytics agent for Linux hasn't picked up the latest configuration.
- The changed settings in the portal weren't applied.

## Resolution

**Background:** `omsconfig` is the Log Analytics agent for Linux configuration agent that looks for new portal-side configuration every five minutes. This configuration is then applied to the Log Analytics agent for Linux configuration files located at `/etc/opt/microsoft/omsagent/conf/omsagent.conf`.

In some cases, the Log Analytics agent for Linux configuration agent might not be able to communicate with the portal configuration service. This scenario results in the latest configuration not being applied.

1. Check that the `omsconfig` agent is installed by running `dpkg --list omsconfig` or `rpm -qi omsconfig`. If it isn't installed, reinstall the latest version of the Log Analytics agent for Linux.
2. Check that the `omsconfig` agent can communicate with Azure Monitor by running the following command: `sudo su omsagent -c 'python /opt/microsoft/omsconfig/Scripts/GetDscConfiguration.py'`. This command returns the configuration that the agent receives from the service, including Syslog settings, Linux performance counters, and custom logs. If this command fails, run the following command:  
`sudo su omsagent -c 'python /opt/microsoft/omsconfig/Scripts/PerformRequiredConfigurationChecks.py'`.  
This command forces the `omsconfig` agent to talk to Azure Monitor and retrieve the latest configuration.

# Issue: You aren't seeing any custom log data

## Probable causes

- Onboarding to Azure Monitor failed.
- The setting **Apply the following configuration to my Linux Servers** hasn't been selected.
- `omsconfig` hasn't picked up the latest custom log configuration from the service.
- The Log Analytics agent for Linux user `omsagent` is unable to access the custom log due to permissions or not being found. You might see the following errors:
  - [DATETIME] [warn]: file not found. Continuing without tailing it.
  - [DATETIME] [error]: file not accessible by omsagent.
- Known issue with race condition fixed in Log Analytics agent for Linux version 1.1.0-217.

## Resolution

1. Verify onboarding to Azure Monitor was successful by checking if the following file exists:  
`/etc/opt/microsoft/omsagent/<workspace id>/conf/omsadmin.conf`. If not, either:
  - a. Reonboard by using the `omsadmin.sh` command line [instructions](#).
  - b. Under **Advanced Settings** in the Azure portal, ensure that the setting **Apply the following configuration to my Linux Servers** is enabled.
2. Check that the `omsconfig` agent can communicate with Azure Monitor by running the following command: `sudo su omsagent -c 'python /opt/microsoft/omsconfig/Scripts/GetDscConfiguration.py'`. This command returns the configuration that the agent receives from the service, including Syslog settings, Linux performance counters, and custom logs. If this command fails, run the following command:  
`sudo su omsagent -c 'python /opt/microsoft/omsconfig/Scripts/PerformRequiredConfigurationChecks.py'`.  
This command forces the `omsconfig` agent to talk to Azure Monitor and retrieve the latest configuration.

**Background:** Instead of the Log Analytics agent for Linux running as a privileged user - `root`, the agent runs as the `omsagent` user. In most cases, explicit permission must be granted to this user for certain files to be read.

To grant permission to `omsagent` user, run the following commands:

1. Add the `omsagent` user to the specific group: `sudo usermod -a -G <GROUPNAME> <USERNAME>`.
2. Grant universal read access to the required file: `sudo chmod -R ugo+rwx <FILE DIRECTORY>`.

There's a known issue with a race condition with the Log Analytics agent for Linux version earlier than 1.1.0-217.

After you update to the latest agent, run the following command to get the latest version of the output plug-in:

```
sudo cp /etc/opt/microsoft/omsagent/sysconf/omsagent.conf /etc/opt/microsoft/omsagent/<workspace id>/conf/omsagent.conf
```

## Issue: You're trying to reonboard to a new workspace

When you try to reonboard an agent to a new workspace, the Log Analytics agent configuration needs to be cleaned up before reonboarding. To clean up old configuration from the agent, run the shell bundle with

```
--purge :
```

```
sudo sh ./omsagent-*universal.x64.sh --purge
```

Or

```
sudo sh ./onboard_agent.sh --purge
```

You can continue to reonboard after you use the `--purge` option.

## Issue: Log Analytics agent extension in the Azure portal is marked with a failed state: Provisioning failed

### Probable causes

- The Log Analytics agent has been removed from the operating system.
- The Log Analytics agent service is down, disabled, or not configured.

### Resolution

1. Remove the extension from the Azure portal.
2. Install the agent by following the [instructions](#).
3. Restart the agent by running the following command:  
`sudo /opt/microsoft/omsagent/bin/service_control restart`.
4. Wait several minutes until the provisioning state changes to **Provisioning succeeded**.

## Issue: The Log Analytics agent upgrade on-demand

### Probable causes

The Log Analytics agent packages on the host are outdated.

### Resolution

1. Check for the latest release on [this GitHub page](#).
2. Download the installation script (1.4.2-124 is an example version):

```
wget https://github.com/Microsoft/OMS-Agent-for-Linux/releases/download/OMSAgent_GA_v1.4.2-124/omsagent-1.4.2-124.universal.x64.sh
```

3. Upgrade packages by executing `sudo sh ./omsagent-*universal.x64.sh --upgrade`.

**Issue:** Installation is failing and says Python2 can't support ctypes, even though Python3 is being used

#### Probable causes

For this known issue, if the VM's language isn't English, a check will fail when verifying which version of Python is being used. This issue leads to the agent always assuming Python2 is being used and failing if there's no Python2.

#### Resolution

Change the VM's environmental language to English:

```
export LANG=en_US.UTF-8
```

# Overview of VM insights

9/21/2022 • 2 minutes to read • [Edit Online](#)

VM insights monitors the performance and health of your virtual machines and virtual machine scale sets, including their running processes and dependencies on other resources. It can help deliver predictable performance and availability of vital applications by identifying performance bottlenecks and network issues and can also help you understand whether an issue is related to other dependencies.

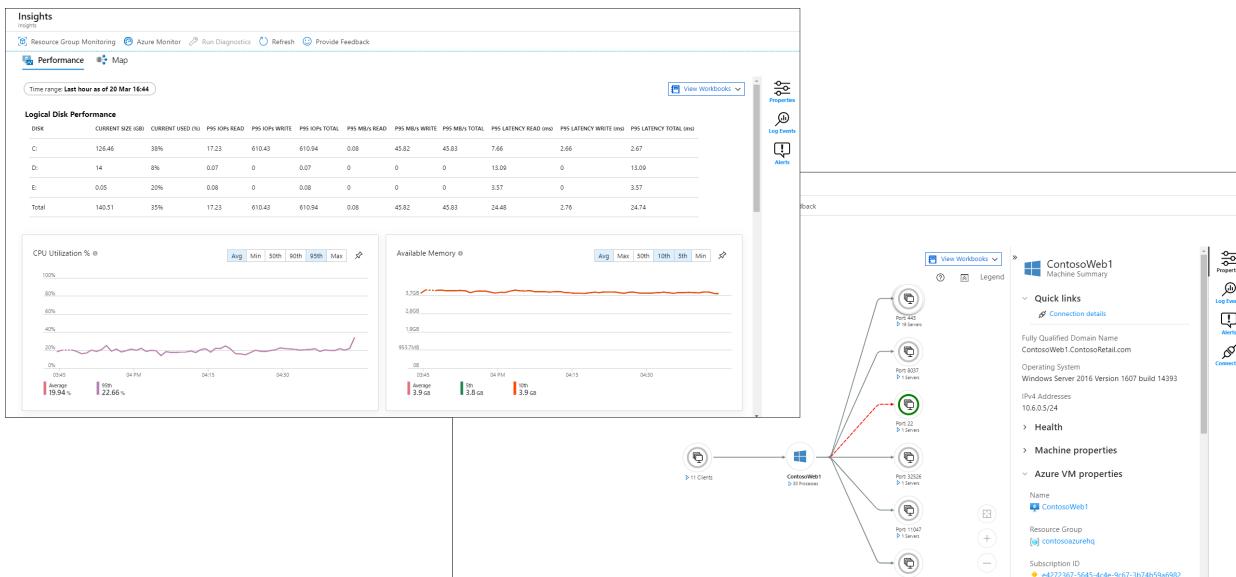
## NOTE

VM insights now supports [Azure Monitor agent](#). See [Enable VM insights overview](#).

VM insights supports Windows and Linux operating systems on the following machines:

- Azure virtual machines
- Azure virtual machine scale sets
- Hybrid virtual machines connected with Azure Arc
- On-premises virtual machines
- Virtual machines hosted in another cloud environment

VM insights stores its data in Azure Monitor Logs, which allows it to deliver powerful aggregation and filtering and to analyze data trends over time. You can view this data in a single VM from the virtual machine directly, or you can use Azure Monitor to deliver an aggregated view of multiple VMs.



## Pricing

There's no direct cost for VM insights, but you're charged for its activity in the Log Analytics workspace. Based on the pricing that's published on the [Azure Monitor pricing page](#), VM insights is billed for:

- Data ingested from agents and stored in the workspace.
- Health state data collected from guest health (preview)
- Alert rules based on log and health data.
- Notifications sent from alert rules.

The log size varies by the string lengths of performance counters, and it can increase with the number of logical disks and network adapters allocated to the VM. If you're already using Service Map, the only change you'll see is the extra performance data that's sent to the Azure Monitor `InsightsMetrics` data type.

## Accessing VM insights

Access VM insights for all your virtual machines and virtual machine scale sets by selecting **Virtual Machines** from the **Monitor** menu in the Azure portal. Access VM insights for a single virtual machine or virtual machine scale set by selecting **Insights** from the machine's menu in the Azure portal.

## Configuring VM insights

The steps to configure VM insights are as follows. Follow each link for detailed guidance on each step:

- [Create Log Analytics workspace](#).
- [Add VMInsights solution to workspace](#). (Log Analytics agent only)
- [Install agents on virtual machine and virtual machine scale set to be monitored](#).

### NOTE

VM Insights does not support sending data to more than one Log Analytics workspace (multi-homing).

## Next steps

- See [Deploy VM insights](#) for requirements and methods that to enable monitoring for your virtual machines.



# Enable VM insights overview

9/21/2022 • 7 minutes to read • [Edit Online](#)

This article provides an overview of the options available to enable VM insights to monitor health and performance of the following:

- Azure virtual machines
- Azure virtual machine scale sets
- Hybrid virtual machines connected with Azure Arc
- On-premises virtual machines
- Virtual machines hosted in another cloud environment.

## Installation options and supported machines

The following table shows the installation methods available for enabling VM insights on supported machines.

METHOD	SCOPE
Azure portal	Enable individual machines with the Azure portal.
Azure Policy	Create policy to automatically enable when a supported machine is created.
Resource Manager templates	Enable multiple machines using any of the supported methods to deploy a Resource Manager template such as CLI and PowerShell.
PowerShell	Use a PowerShell script to enable multiple machines. Log Analytics agent only.
Manual install	Virtual machines or physical computers on-premises other cloud environments. Log Analytics agent only

## Supported Azure Arc machines

VM insights is available for Azure Arc-enabled servers in regions where the Arc extension service is available. You must be running version 0.9 or above of the Arc Agent.

## Supported operating systems

VM insights supports any operating system that supports the Dependency agent and either the Azure Monitor agent (preview) or Log Analytics agent. See [Overview of Azure Monitor agents](#) for a complete list.

### IMPORTANT

If the ethernet device for your virtual machine has more than nine characters, then it won't be recognized by VM insights and data won't be sent to the InsightsMetrics table. The agent will collect data from [other sources](#).

### Linux considerations

See the following list of considerations on Linux support of the Dependency agent that supports VM insights:

- Only default and SMP Linux kernel releases are supported.
- Nonstandard kernel releases, such as Physical Address Extension (PAE) and Xen, aren't supported for any Linux distribution. For example, a system with the release string of *2.6.16.21-0.8-xen* isn't supported.
- Custom kernels, including recompilations of standard kernels, aren't supported.
- For Debian distros other than version 9.4, the map feature isn't supported, and the Performance feature is available only from the Azure Monitor menu. It isn't available directly from the left pane of the Azure VM.
- CentOSPlus kernel is supported.

The Linux kernel must be patched for the Spectre and Meltdown vulnerabilities. Please consult your Linux distribution vendor for more details. Run the following command to check for available if Spectre/Meltdown has been mitigated:

```
$ grep . /sys/devices/system/cpu/vulnerabilities/*
```

Output for this command will look similar to the following and specify whether a machine is vulnerable to either issue. If these files are missing, the machine is unpatched.

```
/sys/devices/system/cpu/vulnerabilities/meltdown:Mitigation: PTI
/sys/devices/system/cpu/vulnerabilities/spectre_v1:Vulnerable
/sys/devices/system/cpu/vulnerabilities/spectre_v2:Vulnerable: Minimal generic ASM retpoline
```

## Log Analytics workspace

VM insights requires a Log Analytics workspace. See [Configure Log Analytics workspace for VM insights](#) for details and requirements of this workspace.

### NOTE

VM Insights does not support sending data to more than one Log Analytics workspace (multi-homing).

## Network requirements

- See [Network requirements](#) for the network requirements for the Log Analytics agent.
- The dependency agent requires a connection from the virtual machine to the address 169.254.169.254. This is the Azure metadata service endpoint. Ensure that firewall settings allow connections to this endpoint.

## Agents

When you enable VM insights for a machine, the following agents are installed. See [Network requirements](#) for the network requirements for these agents.

### IMPORTANT

VM insights support for Azure Monitor agent is currently in public preview. Azure Monitor agent includes several advantages over Log Analytics agent, and is the preferred agent for virtual machines and virtual machine scale sets. See [Migrate to Azure Monitor agent from Log Analytics agent](#) for comparison of the agent and information on migrating.

- [Azure Monitor agent](#) or [Log Analytics agent](#). Collects data from the virtual machine or virtual machine scale set and delivers it to the Log Analytics workspace.
- Dependency agent. Collects discovered data about processes running on the virtual machine and external process dependencies, which are used by the [Map feature in VM insights](#). The Dependency agent relies on

the Azure Monitor agent or Log Analytics agent to deliver its data to Azure Monitor.

## Changes for Azure Monitor agent

There are several changes in the process for enabling VM insights when using the Azure Monitor agent.

**Workspace configuration.** You no longer need to [enable VM insights on the Log Analytics workspace](#) since the VM insights management pack isn't used by Azure Monitor agent.

**Data collection rule.** Azure Monitor agent uses [data collection rules](#) to configure its data collection. VM insights creates a data collection rule that is automatically deployed if you enable your machine using the Azure portal. If you use other methods to onboard your machines, then you may need to install the data collection rule first.

**Agent deployment.** There are minor changes to the process for onboarding virtual machines and virtual machine scale sets to VM insights in the Azure portal. You must now select which agent you want to use, and you must select a data collection rule for Azure Monitor agent. See [Enable VM insights in the Azure portal](#) for details.

## Data collection rule (Azure Monitor agent)

When you enable VM insights on a machine with the Azure Monitor agent you must specify a [data collection rule \(DCR\)](#) to use. The DCR specifies the data to collect and the workspace to use. VM insights creates a default DCR if one doesn't already exist. See [Enable VM insights for Azure Monitor agent](#) for more information on creating and editing the VM insights data collection rule.

### IMPORTANT

It's not recommended to create your own DCR to support VM insights. The DCR created by VM insights includes a special data stream required for its operation. While you can edit this DCR to collect additional data such as Windows and Syslog events, you should create additional DCRs and associate with the machine.

The DCR is defined by the options in the following table.

OPTION	DESCRIPTION
Guest performance	Specifies whether to collect performance data from the guest operating system. This is required for all machines.
Processes and dependencies	Collected details about processes running on the virtual machine and dependencies between machines. This enables the <a href="#">map feature in VM insights</a> . This is optional and enables the <a href="#">VM insights map feature</a> for the machine.
Log Analytics workspace	Workspace to store the data. Only workspaces with VM insights will be listed.

## Management packs (Log Analytics agent)

When a Log Analytics workspace is configured for VM insights, two management packs are forwarded to all the Windows computers connected to that workspace. The management packs are named *Microsoft.IntelligencePacks.ApplicationDependencyMonitor* and *Microsoft.IntelligencePacks.VMInsights* and are written to *%Programfiles%\Microsoft Monitoring Agent\Agent\Health Service State\Management Packs*.

The data source used by the *ApplicationDependencyMonitor* management pack is \*%Program files%\Microsoft

*Monitoring Agent\Agent\Health Service  
State\Resources<AutoGeneratedID>|Microsoft.EnterpriseManagement.Advisor.ApplicationDependencyMonitor  
DataSource.dll. The data source used by the VMInsights management pack is %Program files%\Microsoft  
Monitoring Agent\Agent\Health Service State\Resources<AutoGeneratedID>|  
Microsoft.VirtualMachineMonitoringModule.dll.*

## Migrate from Log Analytics agent

The Azure Monitor agent and the Log Analytics agent can both be installed on the same machine during migration. You should be careful that running both agents may lead to duplication of data and increased cost. If a machine has both agents installed, you'll have a warning in the Azure portal that you may be collecting duplicate data.

### WARNING

Collecting duplicate data from a single machine with both the Azure Monitor agent and Log Analytics agent can result in the following consequences:

- Additional ingestion cost from sending duplicate data to the Log Analytics workspace.
- The map feature of VM insights may be inaccurate since it does not check for duplicate data.

Name	Monitor Coverage	Data collection rule
my-subscription	2 of 20	
srv-01	<span style="color: orange;">!</span> Enabled	MSVMI-mydcr Configure using Azure Monitor Agent
srv-02	Enabled	
my-subscription-01	70 of 87	
my-subscription-02	1 of 1	

You must remove the Log Analytics agent yourself from any machines that are using it. Before you do this, ensure that the machine is not relying any other solutions that require the Log Analytics agent. See [Migrate to Azure Monitor agent from Log Analytics agent](#) for details.

After you verify that no Log Analytics agents are still connected to your Log Analytics workspace, you can [remove the VMInsights solution from the workspace](#) which is no longer needed.

### NOTE

To check if you have any machines with both agents sending data to your Log Analytics workspace, run the following log query in [Log Analytics](#). This will show the last heartbeat for each computer. If a computer has both agents, then it will return two records each with a different category. The Azure Monitor agent will have a category of *Azure Monitor Agent*. The Log Analytics agent will have a category of *Direct Agent*.

```
Heartbeat
| summarize max(TimeGenerated) by Computer, Category
| sort by Computer
```

## Diagnostic and usage data

Microsoft automatically collects usage and performance data through your use of the Azure Monitor service. Microsoft uses this data to improve the quality, security, and integrity of the service.

To provide accurate and efficient troubleshooting capabilities, the Map feature includes data about the configuration of your software. The data provides information such as the operating system and version, IP address, DNS name, and workstation name. Microsoft doesn't collect names, addresses, or other contact information.

For more information about data collection and usage, see the [Microsoft Online Services Privacy Statement](#).

**NOTE**

For information about viewing or deleting personal data, see [Azure Data Subject Requests for the GDPR](#). For more information about GDPR, see the [GDPR section of the Microsoft Trust Center](#) and the [GDPR section of the Service Trust portal](#).

## Next steps

To learn how to use the Performance monitoring feature, see [View VM insights Performance](#). To view discovered application dependencies, see [View VM insights Map](#).

# Enable VM insights in the Azure portal

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article describes how to enable VM insights using the Azure portal for the following :

- Azure virtual machine
- Azure virtual machine scale set
- Hybrid virtual machine connected with Azure Arc

## Prerequisites

- [Create a Log Analytics workspace](#). You can create a new workspace during this process, but you should use an existing workspace if you already have one. See [Log Analytics workspace overview](#) and [Design a Log Analytics workspace architecture](#) for more information.
- See [Supported operating systems](#) to ensure that the operating system of the virtual machine or virtual machine scale set you're enabling is supported.
- See [Manage the Azure Monitor agent](#) for prerequisites related to Azure Monitor agent.

### NOTE

This process describes enabling VM insights from the **Monitor** menu in the Azure portal. You can perform the same process from the **Insights** menu for a particular virtual machine or virtual machine scale set.

## View monitored and unmonitored machines

Open VM insights by selecting **Virtual Machines** from the **Monitor** menu in the Azure portal. The **Overview** page lists all of the virtual machines and virtual machine scale sets in the selected subscriptions. Machines will either be included in the **Monitored** or **Not monitored** tab depending on whether the machine is currently being monitored by VM insights.

A machine may be listed in **Not monitored** even though it has the Azure Monitor or Log Analytics agent installed but has not been enabled for VM insights. If a virtual machine has the Log Analytics agent installed but not the Dependency agent, it will be listed as not monitored. In this case, the Azure Monitor agent will be started without being given the option for the Log Analytics agent.

### NOTE

**Data collection rule** column has replaced the **Workspace** column on the **Overview** page to support the [Azure Monitor agent](#). This either shows the data collection rules used by the Azure Monitor agent for each machine, or it gives the option to configure with the Azure Monitor agent.

## Enable VM insights for Azure Monitor agent

### NOTE

A system-assigned managed identity will be added for a machine as part of the installation process of the Azure Monitor agent if one doesn't already exist.

Use this procedure to enable an unmonitored virtual machine or virtual machine scale set using Azure Monitor agent.

1. Select **Virtual Machines** from the **Monitor** menu in the Azure portal.
2. From the **Overview** page, select **Not Monitored**.
3. Click the **Enable** button next to any machine that you want to enable. If a machine is currently running, then you must start it to enable it.

The screenshot shows the Azure Monitor interface for Virtual Machines. The left sidebar has 'Virtual Machines' selected. The main area shows 'Not monitored (30)' VMs. For 'srv-lx-01', the 'Enable' button is highlighted in blue.

4. Click **Enable** on the introduction page to view the configuration.
5. Select **Azure Monitor agent** from the **Monitoring configuration** page and then select **Azure Monitor agent**.
6. If a **data collection rule (DCR)** hasn't already been created for unmonitored machines, then one will be created with the following details.
  - **Guest performance** enabled.
  - **Processes and dependencies** disabled.
7. If you want this configuration, then click **Configure** to start the agent installation, or select a different data collection rule from the dropdown. Only data collection rules enabled for VM insights will be included.
8. If you want a different configuration or want to use a different Log Analytics workspace, then click **Create new** to create a new data collection rule. This will allow you to select a workspace and specify whether you want to collect processes and dependencies to enable the **map feature in VM insights**.

The screenshot shows the 'Create new rule' dialog. It includes fields for 'Data collection rule name' (set to 'my-dcr'), 'Guest performance' (checkbox checked), 'Processes and dependencies' (checkbox checked), and 'Log Analytics workspaces' (set to 'my-workspace'). At the bottom are 'Configure' and 'Cancel' buttons.

6. Click **Configure** to start the configuration process. It will take several minutes for the agent to be installed and data to start being collected. You'll receive status messages as the configuration is performed.
7. If you use a manual upgrade model for your virtual machine scale set, upgrade the instances to complete the setup. You can start the upgrades from the **Instances** page, in the **Settings** section.

## Enable VM insights for Log Analytics agent

Use this procedure to enable an unmonitored virtual machine or virtual machine scale set using Log Analytics agent.

1. Select **Virtual Machines** from the **Monitor** menu in the Azure portal.
2. From the **Overview** page, select **Not Monitored**.
3. Click the **Enable** button next to any machine that you want to enable. If a machine is currently running, then you must start it to enable it.

Name	Monitor Coverage	Workspace
my-vm	30 of 36	
srv-lx-01	Not enabled	<b>Enable</b>
srv-lx-02	Cannot enable - Virtual machine is not running (Why?)	bren

3. Click **Enable** on the introduction page to view the configuration.
4. Select **Azure Monitor agent** from the **Monitoring configuration** page and then select **Log Analytics agent**.
5. If the virtual machine isn't already connected to a Log Analytics workspace, then you'll be prompted to select one. If you haven't previously [created a workspace](#), then you can select a default for the location where the virtual machine or virtual machine scale set is deployed in the subscription. This workspace will be created and configured if it doesn't already exist. If you select an existing workspace, it will be configured for VM insights if it wasn't already.

### NOTE

If you select a workspace that wasn't previously configured for VM insights, the *VMInsights* management pack will be added to this workspace. This will be applied to any agent already connected to the workspace, whether or not it's enabled for VM insights. Performance data will be collected from these virtual machines and stored in the *InsightsMetrics* table.

6. Click **Configure** to modify the configuration. The only option you can modify is the workspace. You will receive status messages as the configuration is performed.
7. If you use a manual upgrade model for your virtual machine scale set, upgrade the instances to complete the setup. You can start the upgrades from the **Instances** page, in the **Settings** section.

## Enable Azure Monitor agent on monitored machines

Use this procedure to add the Azure Monitor agent to machines that are already enabled with the Log Analytics agent.

1. Select **Virtual Machines** from the **Monitor** menu in the Azure portal.
2. From the **Overview** page, select **Monitored**.
3. Click **Configure using Azure Monitor agent** next to any machine that you want to enable. If a machine is currently running, then you must start it to enable it.

The screenshot shows the Azure portal's 'Monitored' section. At the top, there are filters for Subscription (43 subscriptions), Resource group (All resource groups), Type (All types), and Location (All locations). Below the filters, there are tabs for 'Monitored' (73), 'Not monitored' (45), 'Workspace configuration', and 'Other onboarding options'. The main table lists resources by name, monitor coverage, and data collection rule. A red box highlights the 'Configure using Azure Monitor Agent' button for the 'srv-02' machine under the 'my-resource-group' entry.

4. Follow the process described in [Enable VM insights for Azure Monitor agent](#) to select a data collection rule. The only difference is that the data collection rule hasn't created for monitored machines has **Processes and dependencies** enabled for backward compatibility with the Log Analytics agent.

The screenshot shows the 'Monitoring configuration' dialog. On the left, the Azure portal navigation bar is visible with 'Monitor' selected. The main area shows a list of resources under 'my-subscription' and 'my-workspace'. On the right, the 'Monitoring configuration' dialog is open, showing the 'Data collection rule' dropdown set to 'MSVMI-my-workspace'. The 'Processes and dependencies' checkbox is checked. A note at the bottom states: 'This will also enable System Assigned Managed Identity, in addition to existing User Assigned identities (if any). Note: Unless specified in the request, the machine will default to using System Assigned Identity. Learn More'. Buttons for 'Configure' and 'Cancel' are at the bottom.

5. With both agents installed, a warning will be displayed indicating that you may be collecting duplicate data.

Get Started   Performance   Map

Filter by name...   Subscription : 43 subscriptions   Resource group : All resource groups   Type : All types   Location : All locations

Monitored (73)   Not monitored (45)   Workspace configuration   Other onboarding options

Name	Monitor Coverage	Data collection rule
my-subscription	2 of 20	
srv-01	<span style="color: orange;">⚠ Enabled</span>	MSVMI-mydcr
srv-02	Enabled	Configure using Azure Monitor Agent
my-subscription-01	70 of 87	
my-subscription-02	1 of 1	

The virtual machine is onboarded via Microsoft monitoring agent as well as Azure Monitoring agent resulting in duplication of data and increase cost.

### WARNING

Collecting duplicate data from a single machine with both the Azure Monitor agent and Log Analytics agent can result in the following consequences:

- Additional ingestion cost from sending duplicate data to the Log Analytics workspace.
- The map feature of VM insights may be inaccurate since it does not check for duplicate data.

See [Migrate from Log Analytics agent](#).

6. Once you've verified that the Azure Monitor agent has been enabled, remove the Log Analytics agent from the machine to prevent duplicate data collection.

## Next steps

- See [Use VM insights Map](#) to view discovered application dependencies.
- See [View Azure VM performance](#) to identify bottlenecks, overall utilization, and your VM's performance.

# Enable VM insights by using Azure Policy

9/21/2022 • 7 minutes to read • [Edit Online](#)

This article explains how to enable VM insights for Azure virtual machines or hybrid virtual machine connected with Azure Arc (preview) using Azure Policy. Azure Policy allows you to assign policy definitions that install the required agents for VM insights across your Azure environment and automatically enable monitoring for VMs as each virtual machine is created. VM insights provides a feature that allows you to discover and remediate noncompliant VMs in your environment. Use this feature instead of working directly with Azure Policy.

If you're not familiar with Azure Policy, get a brief introduction at [Deploy Azure Monitor at scale using Azure Policy](#).

## NOTE

This article describes VM insights using the Log Analytics agent. VM insights with the Azure Monitor agent is currently in public preview. See [Enable VM insights overview](#) for details on installing with this agent.

## NOTE

To use Azure Policy with Azure virtual machine scale sets, or to work with Azure Policy directly to enable Azure virtual machines, see [Deploy Azure Monitor at scale using Azure Policy](#).

## VM insights initiatives

VM insights provides builtin policy definitions to install the Log Analytics agent and Dependency agent on Azure virtual machines. The following built-in initiatives install both agents to enable full monitoring. Assign these initiatives to a management group, subscription, or resource group to automatically install the agents on any Windows or Linux Azure virtual machines in that scope.

NAME	DESCRIPTION
Enable VM insights	Installs the Log Analytics agent and Dependency agent on Azure VMs and hybrid VMs connected with Azure Arc.
Enable Azure Monitor for virtual machine scale sets	Installs the Log Analytics agent and Dependency agent on Azure virtual machine scale sets.

## Open Policy Coverage feature

To access **VM insights Policy Coverage**, go the **Virtual machines** in the **Azure Monitor** menu in the Azure portal. Select **Other onboarding options** and then **Enable** under **Enable using policy**.

Monitor | Virtual Machines

Get Started

15 Monitored 114 Not monitored Workspace configuration Other onboarding options

Enable for a single VM

Use the 'Not Monitored' tab to identify and enable insights for Azure VMs. You will need contributor access on the VM, and Log Analytics contributor access on the Resource Group of your chosen Log Analytics workspace.

Learn more

Enable

Enable using policy

Use Azure Policy to ensure all VMs and VM Scale Sets in your subscriptions and resource groups are configured for monitoring. You will need role access as Owner or User Access Administrator for the selected subscriptions.

Learn more

Enable

Configure a workspace

Use this to enable Azure Monitor for VMs features on Log Analytics workspace.

Learn more

Configure

## Create new assignment

If you don't already have an assignment, create a new one by clicking **Assign Policy**.

Home > Monitor | Virtual Machines >

### Azure Monitor for VMs Policy Coverage

Initiative: Enable Azure Monitor for VMs

Refresh Configure Workspace Provide Feedback

Search by name or ID

Assess your management groups and subscriptions that do not have the Azure Monitor for VMs agent deployment initiative assigned.

Learn more

Tenant Root Group <small>(details)</small>							
Scope	My Role	Total VMs	Assignment Cover...	Assignment Status	Compliant VMs	Compliance	Compliance State
My Mgmt Group	Owner	3	0%	0%	0	0	0

Assign Policy

This is the same page to assign an initiative in Azure Policy except that it's hardcoded with the scope that you selected and the **Enable VM insights** initiative definition. You can optionally change the **Assignment name** and add a **Description**. Select **Exclusions** if you want to provide an exclusion to the scope. For example, your scope could be a management group, and you could specify a subscription in that management group to be excluded from the assignment.

Home > Monitor | Virtual Machines > Azure Monitor for VMs Policy Coverage >

## Enable Azure Monitor for VMs

Assign initiative

**Basics**   Parameters   Remediation   Review + create

**Scope**  
Scope [Learn more about setting the scope \\*](#)

My Mgmt Group

**Exclusions**  
Optional: select resources to exclude from the policy assignment.

**Basics**  
Initiative definition  
Enable Azure Monitor for VMs

**Assignment name \*** [\(i\)](#)  
Enable Azure Monitor for VMs

**Description**

**Policy enforcement** [\(i\)](#)  
 Enabled  Disabled

**Assigned by**  
Administrator

**Review + create**

On the **Parameters** page, select a **Log Analytics workspace** to be used by all virtual machines in the assignment. If you want to specify different workspaces for different virtual machines, then you must create multiple assignments, each with their own scope.

### NOTE

If the workspace is beyond the scope of the assignment, grant *Log Analytics Contributor* permissions to the policy assignment's Principal ID. If you don't do this, you might see a deployment failure like

The client '343de0fe-e724-46b8-b1fb-97090f7054ed' with object id '343de0fe-e724-46b8-b1fb-97090f7054ed' does not have authorization to perform action 'microsoft.operationalinsights/workspaces/read' over scope ...

Home > Monitor | Virtual Machines > Azure Monitor for VMs Policy Coverage >

## Enable Azure Monitor for VMs

Assign initiative

**Basics**   **Parameters**   Remediation   Review + create

Specify parameters for this initiative assignment.

**Log Analytics workspace \*** [\(i\)](#)  
my-workspace

Optional: List of VM images that have supported Windows OS to add to scope [\(i\)](#)

Optional: List of VM images that have supported Linux OS to add to scope [\(i\)](#)

**Review + create**

Click **Review + Create** to review the details of the assignment before clicking **Create** to create it. Don't create a remediation task at this point since you will most likely need multiple remediation tasks to enable existing virtual machines. See [Remediate compliance results](#) below.

### Review compliance

Once an assignment is created, you can review and manage coverage for the **Enable VM insights** initiative across your management groups and subscriptions. This will show how many virtual machines exist in each of the management groups or subscriptions and their compliance status.

The screenshot shows the Azure Monitor for VMs Policy Coverage initiative page. At the top, there's a search bar labeled "Search by name or ID". Below it, a section titled "Tenant Root Group (details)" shows a single entry: "IT Organization Mgmt Group" with "Owner" role, "16" total VMs, and "100%" assignment coverage, marked as "Complete". To the right, a summary table provides overall compliance statistics: 5 compliant VMs (31%) and 1 non-compliant VM (69%).

The following table provides a description of the information in this view.

FUNCTION	DESCRIPTION
Scope	Management group and subscriptions that you have or inherited access to with ability to drill down through the management group hierarchy.
Role	Your role in the scope, which might be reader, owner, or contributor. This will be blank if you have access to the subscription but not to the management group it belongs to. This role determines what data you can see and actions you can perform in terms of assigning policies or initiatives (owner), editing them, or viewing compliance.
Total VMs	Total number of VMs in that scope regardless of their status. For a management group, this is a sum total of VMs nested under the subscriptions or child management groups.
Assignment Coverage	Percent of VMs that are covered by the initiative.
Assignment Status	<p><b>Success</b> - All VMs in the scope have the Log Analytics and Dependency agents deployed to them.</p> <p><b>Warning</b> - The subscription isn't under a management group.</p> <p><b>Not Started</b> - A new assignment was added.</p> <p><b>Lock</b> - You don't have sufficient privileges to the management group.</p> <p><b>Blank</b> - No VMs exist or a policy isn't assigned.</p>
Compliant VMs	Number of VMs that are compliant, which is the number of VMs that have both Log Analytics agent and Dependency agent installed. This will be blank if there are no assignments, no VMs in the scope, or not proper permissions.
Compliance	The overall compliance number is the sum of distinct resources that are compliant divided by the sum of all distinct resources.
Compliance State	<p><b>Compliant</b> - All VMs in the scope virtual machines have the Log Analytics and Dependency agents deployed to them or any new VMs in the scope subject to the assignment have not yet been evaluated.</p> <p><b>Non-compliant</b> - There are VMs that have been evaluated but are not enabled and may require remediation.</p> <p><b>Not Started</b> - A new assignment was added.</p> <p><b>Lock</b> - You don't have sufficient privileges to the management group.</p> <p><b>Blank</b> - No policy is assigned.</p>

When you assign the initiative, the scope selected in the assignment could be the scope listed or a subset of it. For instance, you might have created an assignment for a subscription (policy scope) and not a management group (coverage scope). In this case, the value of **Assignment Coverage** indicates the VMs in the initiative scope divided by the VMs in coverage scope. In another case, you might have excluded some VMs, resource groups, or a subscription from policy scope. If the value is blank, it indicates that either the policy or initiative doesn't exist or you don't have permission. Information is provided under **Assignment Status**.

## Remediate compliance results

The initiative will be applied to virtual machines as they're created or modified, but it won't be applied to existing VMs. If your assignment doesn't show 100% compliance, create remediation tasks to evaluate and enable existing VMs, select **View Compliance** by selecting the ellipsis (...).

The screenshot shows the 'Azure Monitor for VMs Policy Coverage' page. At the top, there's a search bar and navigation links for Refresh, Configure Workspace, and Provide Feedback. Below that, a message says 'Assess your management groups and subscriptions that do not have the Azure Monitor for VMs agent deployment initiative assigned.' A 'Learn more' link is provided. The main table has columns: Scope, My Role, Total VMs, Assignment Covera..., Assignment Status, Compliant VMs, Compliance, and Compliance State. One row shows 'My Mgmt Group' as the scope, 'Owner' as the role, 5 total VMs, 100% coverage (green), 'Complete' status, 4 compliant VMs, 80% compliance, and 'Non-compliant' state. A context menu is open over this row, listing 'Assign Policy', 'Edit Assignment', 'View Compliance' (which is highlighted with a red box), and 'Learn more'. There's also a '...' option at the end of the menu.

The **Compliance** page lists assignments matching the specified filter and whether they're compliant. Click on an assignment to view its details.

The screenshot shows the 'Policy | Compliance' page. On the left, there's a sidebar with links like Home, Overview, Getting started, Join Preview, Compliance (which is selected and highlighted in grey), Remediation, Authoring, Assignments (which is selected and highlighted in grey), Definitions, Related Services, Blueprints (preview), and Resource Graph. The main area has sections for Overall resource compliance (80%, 4 out of 5), Non-compliant initiatives (1, 1 out of 1), Non-compliant policies (1, 1 out of 10), and Non-compliant resources (1, 1 out of 5). Below these is a table for 'Non-Compliant Resource Details'. The first row shows an assignment named 'Enable Azure Monitor for VMs' with scope 'My Mgmt Group', status 'Non-compliant', and compliance '80% (4 out of 5)'. A context menu is open over this row, with the 'View Compliance' option highlighted with a red box. There are also 'Assign policy', 'Edit assignment', and 'Learn more' options in the menu.

The **Initiative compliance** page lists the policy definitions in the initiative and whether each is in compliance.

Home > Enable Azure Monitor for VMs

Initiative compliance

Compliance state: Non-compliant (80% out of 5)

Overall resource compliance: 80% (4 out of 5)

Non-compliant policies: 1 out of 10

Non-compliant resources: 1 out of 5

Events (last 7 days): Audit 0, Append 0, Modify 0, Deny 0, Deploy 0

Policies, Non-compliant resources, Events, Remediation tasks, Deployed Resources

Name	Effect Type	Compliance state	Non-Compliant Resources	Total resources
[Preview]: Audit Log Analytics Agent Deployment - VM Image (OS) unlisted	AuditIfNotExists	Compliant	0	0
[Preview]: Deploy Dependency agent to hybrid Linux Azure Arc machines	DeployIfNotExists	Compliant	0	0
[Preview]: Deploy Dependency agent to Windows Azure Arc machines	DeployIfNotExists	Compliant	0	0
[Preview]: Deploy Log Analytics agent to Linux Azure Arc machines	DeployIfNotExists	Compliant	0	0
[Preview]: Deploy Log Analytics agent to Windows Azure Arc machines	DeployIfNotExists	Compliant	0	0
Audit Dependency agent deployment - VM Image (OS) unlisted	AuditIfNotExists	Compliant	0	0
<b>Deploy Dependency agent for Linux virtual machines</b>	DeployIfNotExists	Non-compliant	1	2
Deploy Dependency agent for Windows virtual machines	DeployIfNotExists	Compliant	0	3
Deploy Log Analytics agent for Linux VMs	DeployIfNotExists	Compliant	0	2
Deploy Log Analytics agent for Windows VMs	DeployIfNotExists	Compliant	0	3

Click on a policy definition to view its details. Scenarios that policy definitions will show as out of compliance include the following:

- Log Analytics agent or Dependency agent isn't deployed. Create a remediation task to mitigate.
- VM image (OS) isn't identified in the policy definition. The criteria of the deployment policy include only VMs that are deployed from well-known Azure VM images. Check the documentation to see whether the VM OS is supported.
- VMs aren't logging to the specified Log Analytics workspace. Some VMs in the initiative scope are connected to a Log Analytics workspace other than the one that's specified in the policy assignment.

Home > Enable Azure Monitor for VMs > Deploy Dependency agent for Linux virtual machines

Policy compliance

Name: Deploy Dependency agent for Linux virtual machines

Description: --

Assignment ID: /providers/Microsoft.Management/managementGroups/bwren-lab/providers/Microsoft...

Scope: My Mgmt Group

Excluded scopes: 0

Definition: Deploy Dependency agent for Linux virtual machines

Selected Scopes: 1 selected management group

Compliance state: Non-compliant (50% out of 2)

Overall resource compliance: 50% (1 out of 2)

Non-compliant resources: 1 out of 2

Events (last 7 days): Audit 0, Append 0, Modify 0, Deny 0, Deploy 0

Details: Effect type: DeployIfNotExists, Parent Initiative: Enable Azure Monitor for VMs

Resource compliance, Events, Remediation tasks, Deployed Resources

Name	Compliance state	Compliance reason	Resource Type	Location	Scope	Last evaluated
my-vm-01	Non-compliant	Details	Microsoft.Compute/...	--	Visual Studio Enterprise with MSDN/bwre...	7/6/2020, 1:25 PM
my-vm-02	Non-compliant	Details	Microsoft.Compute/...	--	Visual Studio Enterprise with MSDN/bwre...	7/6/2020, 1:25 PM

To create a remediation task to mitigate compliance issues, click **Create Remediation Task**.

Home > Enable Azure Monitor for VMs > Deploy Dependency agent for Linux virtual machines >

## New remediation task

**REMEDIATION ACTION**

Policy to remediate ⓘ  
Deploy Dependency agent for Linux virtual machines

**View definition**

Description:  
Deploy Dependency agent for Linux virtual machines if the VM Image (OS) is in the list defined and the agent is not installed.

**RESOURCES TO REMEDIATE**

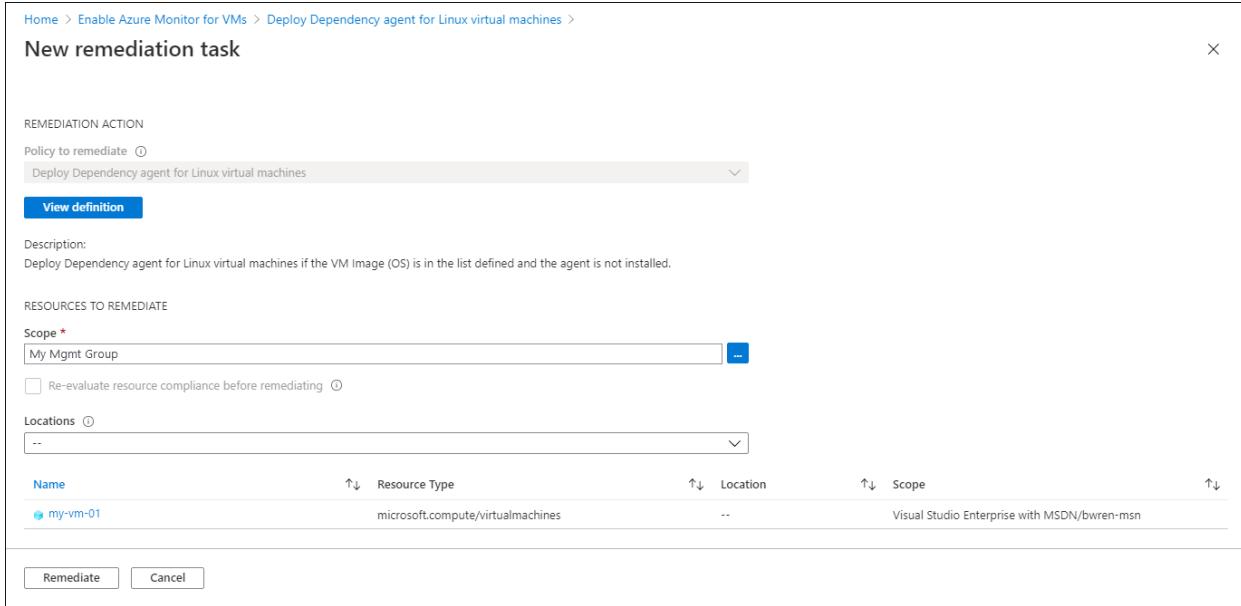
Scope \*  
My Mgmt Group

Re-evaluate resource compliance before remediating ⓘ

Locations ⓘ  
--

Name	Resource Type	Location	Scope
my-vm-01	microsoft.compute/virtualmachines	--	Visual Studio Enterprise with MSDN/bwren-msn

**Remediate** **Cancel**



Click **Remediate** to create the remediation task and then **Remediate** to start it. You will most likely need to create multiple remediation tasks, one for each policy definition. You can't create a remediation task for an initiative.

Home > Monitor | Virtual Machines > Azure Monitor for VMs Policy Coverage >

## Policy | Remediation

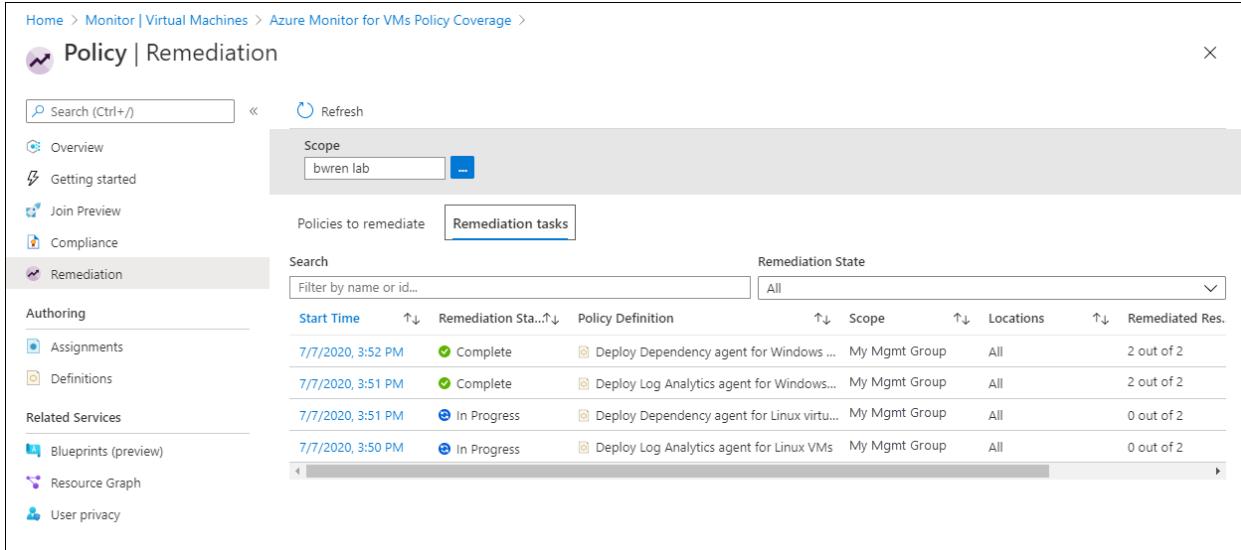
Search (Ctrl+ /) Refresh

Scope  
bwren lab

Policies to remediate Remediation tasks

Search  
Filter by name or id... Remediation State  
All

Start Time	Remediation Sta...	Policy Definition	Scope	Locations	Remediated Res.
7/7/2020, 3:52 PM	Complete	Deploy Dependency agent for Windows ...	My Mgmt Group	All	2 out of 2
7/7/2020, 3:51 PM	Complete	Deploy Log Analytics agent for Windows...	My Mgmt Group	All	2 out of 2
7/7/2020, 3:51 PM	In Progress	Deploy Dependency agent for Linux virtu...	My Mgmt Group	All	0 out of 2
7/7/2020, 3:50 PM	In Progress	Deploy Log Analytics agent for Linux VMs	My Mgmt Group	All	0 out of 2



Once the remediation tasks are complete, your VMs should be compliant with agents installed and enabled for VM insights.

## Azure Policy

To use Azure Policy to enable monitoring for virtual machine scale sets, assign the **Enable Azure Monitor for Virtual Machine Scale Sets** initiative to an Azure management group, subscription, or resource group, depending on the scope of your resources to monitor. A **management group** is useful for scoping policy, especially if your organization has multiple subscriptions.

## Assign initiative

Basics Parameters Remediation Review + create

### Scope

Scope [Learn more about setting the scope \\*](#)

 ...

### Exclusions

Optionally select resources to exclude from the policy assignment.

### Basics

Initiative definition \*

 ...

Assignment name \* ⓘ

 ...

### Description

Policy enforcement ⓘ

Enabled  Disabled

[Review + create](#)

[Cancel](#)

[Previous](#)

[Next](#)

Select the workspace that the data will be sent to. This workspace must have the *VMInsights* solution installed, as described in [Configure Log Analytics workspace for VM insights](#).

## Assign initiative

Basics    **Parameters**    Remediation    Review + create

Specify parameters for this initiative assignment.

Log Analytics workspace \* ⓘ

my-workspace ...

Optional: List of VM images that have supported Windows OS to add to scope ⓘ

Optional: List of VM images that have supported Linux OS to add to scope ⓘ

**Review + create**

Cancel

Previous

Next

Create a remediation task if you have existing virtual machine scale sets that need to be assigned this policy.

## Assign initiative

Basics Parameters Remediation Review + create

By default, this assignment will only take effect on newly created resources. Existing resources can be updated via a remediation task after the policy is assigned. For deployIfNotExists policies, the remediation task will deploy the specified template. For modify policies, the remediation task will edit tags on the existing resources.

Create a remediation task ⓘ

Policy to remediate

Deploy Log Analytics agent for Windows virtual machine scale sets



### Managed Identity

Policies with the deployIfNotExists and modify effect types need the ability to deploy resources and edit tags on existing resources respectively. To do this, a managed identity will be created for you.

[Learn more about Managed Identity.](#)

Create a Managed Identity ⓘ

Managed identity location \*

East US



### Permissions

This identity will also be given the following permissions:

Log Analytics Contributor, Virtual Machine Contributor



i Role assignments (permissions) are created based on the role definitions specified in the policies.

[Review + create](#)

[Cancel](#)

[Previous](#)

[Next](#)

## Next steps

Now that monitoring is enabled for your virtual machines, this information is available for analysis with VM insights.

- To view discovered application dependencies, see [View VM insights Map](#).
- To identify bottlenecks and overall utilization with your VM's performance, see [View Azure VM performance](#).

# Enable VM insights using PowerShell

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article describes how to enable VM insights on Azure virtual machines using PowerShell. This procedure can be used for the following:

- Azure virtual machine
- Azure virtual machine scale set

## NOTE

This article only applies to the Log Analytics agent. To enable VM insights with the Azure monitor agent, use other installation methods described in [Enable VM insights overview](#).

## Prerequisites

- [Create and configure a Log Analytics workspace](#).
- See [Supported operating systems](#) to ensure that the operating system of the virtual machine or virtual machine scale set you're enabling is supported.

## PowerShell script

To enable VM insights for multiple VMs or virtual machine scale sets, use the PowerShell script [Install-VMInsights.ps1](#), which is available from the Azure PowerShell Gallery. This script iterates through:

- Every virtual machine and virtual machine scale set in your subscription.
- The scoped resource group that's specified by *ResourceGroup*.
- A single VM or virtual machine scale set that's specified by *Name*.

For each virtual machine or virtual machine scale set, the script verifies whether the VM extension for the Log Analytics agent and Dependency agent is already installed. If both extensions are installed, the script tries to reinstall it. If both extensions aren't installed, the script installs them.

Verify you are using Azure PowerShell module Az version 1.0.0 or later with `Enable-AzModule` compatibility aliases enabled. Run `Get-Module -ListAvailable Az` to find the version. If you need to upgrade, see [Install Azure PowerShell module](#). If you're running PowerShell locally, you also need to run `Connect-AzAccount` to create a connection with Azure.

To get a list of the script's argument details and example usage, run `Get-Help`.

```
Get-Help .\Install-VMInsights.ps1 -Detailed
```

### SYNOPSIS

This script installs VM extensions for Log Analytics and the Dependency agent as needed for VM Insights.

### SYNTAX

```
.\Install-VMInsights.ps1 [-WorkspaceId] <String> [-WorkspaceKey] <String> [-SubscriptionId] <String> [[-ResourceGroup] <String>] [[-Name] <String>] [[-PolicyAssignmentName] <String>] [-ReInstall] [-TriggerVmssManualVMUpdate] [-Approve] [-WorkspaceRegion] <String> [-WhatIf] [-Confirm] [<CommonParameters>]
```

## DESCRIPTION

This script installs or reconfigures the following on VMs and virtual machine scale sets:

- Log Analytics VM extension configured to supplied Log Analytics workspace
- Dependency agent VM extension

Can be applied to:

- Subscription
- Resource group in a subscription
- Specific VM or virtual machine scale set
- Compliance results of a policy for a VM or VM extension

Script will show you a list of VMs or virtual machine scale sets that will apply to and let you confirm to continue.

Use -Approve switch to run without prompting, if all required parameters are provided.

If the extensions are already installed, they will not install again.

Use -ReInstall switch if you need to, for example, update the workspace.

Use -WhatIf if you want to see what would happen in terms of installs, what workspace configured to, and status of the extension.

## PARAMETERS

-WorkspaceId <String>

Log Analytics WorkspaceID (GUID) for the data to be sent to

-WorkspaceKey <String>

Log Analytics Workspace primary or secondary key

-SubscriptionId <String>

SubscriptionId for the VMs/VM Scale Sets

If using PolicyAssignmentName parameter, subscription that VMs are in

-ResourceGroup <String>

<Optional> Resource Group to which the VMs or VM Scale Sets belong

-Name <String>

<Optional> To install to a single VM/VM Scale Set

-PolicyAssignmentName <String>

<Optional> Take the input VMs to operate on as the Compliance results from this Assignment

If specified will only take from this source.

-ReInstall [<SwitchParameter>]

<Optional> If VM/VM Scale Set is already configured for a different workspace, set this to change to the new workspace

-TriggerVmssManualVMUpdate [<SwitchParameter>]

<Optional> Set this flag to trigger update of VM instances in a scale set whose upgrade policy is set to Manual

-Approve [<SwitchParameter>]

<Optional> Gives the approval for the installation to start with no confirmation prompt for the listed VMs/VM Scale Sets

-WorkspaceRegion <String>

Region the Log Analytics Workspace is in

Supported values: "East US","eastus","Southeast Asia","southeastasia","West Central US","westcentralus","West Europe","westeurope"

For Health supported is: "East US","eastus","West Central US","westcentralus"

-WhatIf [<SwitchParameter>]

<Optional> See what would happen in terms of installs.

If extension is already installed will show what workspace is currently configured, and status of the VM extension

-Confirm [<SwitchParameter>]

<Optional> Confirm every action

```
<CommonParameters>
    This cmdlet supports the common parameters: Verbose, Debug,
    ErrorAction, ErrorVariable, WarningAction, WarningVariable,
    OutBuffer, PipelineVariable, and OutVariable. For more information, see
    about_CommonParameters (https://go.microsoft.com/fwlink/?LinkID=113216).

----- EXAMPLE 1 -----
.\Install-VMInsights.ps1 -WorkspaceRegion eastus -WorkspaceId <WorkspaceId> -WorkspaceKey <WorkspaceKey>
-SubscriptionId <SubscriptionId>
-ResourceGroup <ResourceGroup>

Install for all VMs in a resource group in a subscription

----- EXAMPLE 2 -----
.\Install-VMInsights.ps1 -WorkspaceRegion eastus -WorkspaceId <WorkspaceId> -WorkspaceKey <WorkspaceKey>
-SubscriptionId <SubscriptionId>
-ResourceGroup <ResourceGroup> -ReInstall

Specify to reinstall extensions even if already installed, for example, to update to a different
workspace

----- EXAMPLE 3 -----
.\Install-VMInsights.ps1 -WorkspaceRegion eastus -WorkspaceId <WorkspaceId> -WorkspaceKey <WorkspaceKey>
-SubscriptionId <SubscriptionId>
-PolicyAssignmentName a4f79f8ce891455198c08736 -ReInstall

Specify to use a PolicyAssignmentName for source and to reinstall (move to a new workspace)
```

The following example demonstrates using the PowerShell commands in the folder to enable VM insights and understand the expected output:

```

$WorkspaceId = "<GUID>"
$WorkspaceKey = "<Key>"
$SubscriptionId = "<GUID>"
.\Install-VMInsights.ps1 -WorkspaceId $WorkspaceId -WorkspaceKey $WorkspaceKey -SubscriptionId
$SubscriptionId -WorkspaceRegion eastus

Getting list of VMs or virtual machine scale sets matching criteria specified

VMs or virtual machine scale sets matching criteria:

db-ws-1 VM running
db-ws2012 VM running

This operation will install the Log Analytics and Dependency agent extensions on the previous two VMs or
virtual machine scale sets.
VMs in a non-running state will be skipped.
Extension will not be reinstalled if already installed. Use -ReInstall if desired, for example, to update
workspace.

Confirm
Continue?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y

db-ws-1 : Deploying DependencyAgentWindows with name DAExtension
db-ws-1 : Successfully deployed DependencyAgentWindows
db-ws-1 : Deploying MicrosoftMonitoringAgent with name MMAExtension
db-ws-1 : Successfully deployed MicrosoftMonitoringAgent
db-ws2012 : Deploying DependencyAgentWindows with name DAExtension
db-ws2012 : Successfully deployed DependencyAgentWindows
db-ws2012 : Deploying MicrosoftMonitoringAgent with name MMAExtension
db-ws2012 : Successfully deployed MicrosoftMonitoringAgent

Summary:

Already onboarded: (0)

Succeeded: (4)
db-ws-1 : Successfully deployed DependencyAgentWindows
db-ws-1 : Successfully deployed MicrosoftMonitoringAgent
db-ws2012 : Successfully deployed DependencyAgentWindows
db-ws2012 : Successfully deployed MicrosoftMonitoringAgent

Connected to different workspace: (0)

Not running - start VM to configure: (0)

Failed: (0)

```

## Next steps

- See [Use VM insights Map](#) to view discovered application dependencies.
- See [View Azure VM performance](#) to identify bottlenecks, overall utilization, and your VM's performance.

# Enable VM insights for a hybrid virtual machine

9/21/2022 • 5 minutes to read • [Edit Online](#)

This article describes how to enable VM insights for a virtual machine outside of Azure, including on-premises and other cloud environments.

## IMPORTANT

The recommended method of enabling hybrid VMs is first enabling [Azure Arc for servers](#) so that the VMs can be enabled for VM insights using processes similar to Azure VMs. This article describes how to onboard hybrid VMs if you choose not to use Azure Arc.

## NOTE

This article describes VM insights using the Log Analytics agent. VM insights with the Azure Monitor agent is currently in public preview. See [Enable VM insights overview](#) for details on installing with this agent.

## Prerequisites

- [Create and configure a Log Analytics workspace](#).
- See [Supported operating systems](#) to ensure that the operating system of the virtual machine or virtual machine scale set you're enabling is supported.

## Overview

Virtual machines outside of Azure require the same Log Analytics agent and Dependency agent that are used for Azure VMs. Since you can't use VM extensions to install the agents though, you must manually install them in the guest operating system or have them installed through some other method.

See [Connect Windows computers to Azure Monitor](#) or [Connect Linux computers to Azure Monitor](#) for details on deploying the Log Analytics agent. Details for the Dependency agent are provided in this article.

## Firewall requirements

Firewall requirements for the Log Analytics agent are provided in [Log Analytics agent overview](#). The VM insights Map Dependency agent doesn't transmit any data itself, and it doesn't require any changes to firewalls or ports. The Map data is always transmitted by the Log Analytics agent to the Azure Monitor service, either directly or through the [Operations Management Suite gateway](#) if your IT security policies don't allow computers on the network to connect to the internet.

## Dependency agent

### NOTE

The following information described in this section is also applicable to the [Service Map solution](#).

You can download the Dependency agent from these locations:

FILE	OS	VERSION	SHA-256
<a href="#">InstallDependencyAgent-Windows.exe</a>	Windows	9.10.14.20760	D4DB398FAD36E86FEACCC 41D7B8AF46711346A9438 06769B6CE017F0BF1625FF
<a href="#">InstallDependencyAgent-Linux64.bin</a>	Linux	9.10.14.20760	3DE3B485BA79B57E74B3D FB60FD277A30C8A5D1BD 898455AD77FECF20E0E26 10

## Install the Dependency agent on Windows

You can install the Dependency agent manually on Windows computers by running

`InstallDependencyAgent-Windows.exe`. If you run this executable file without any options, it starts a setup wizard that you can follow to install the agent interactively. You require *Administrator* privileges on the guest OS to install or uninstall the agent.

The following table highlights the parameters that are supported by setup for the agent from the command line.

PARAMETER	DESCRIPTION
<code>/?</code>	Returns a list of the command-line options.
<code>/S</code>	Performs a silent installation with no user interaction.

For example, to run the installation program with the `/?` parameter, enter `InstallDependencyAgent-Windows.exe /?`.

Files for the Windows Dependency agent are installed in `C:\Program Files\Microsoft Dependency Agent` by default. If the Dependency agent fails to start after setup is finished, check the logs for detailed error information. The log directory is `%Programfiles%\Microsoft Dependency Agent\logs`.

### PowerShell script

Use the following sample PowerShell script to download and install the agent:

```
Invoke-WebRequest "https://aka.ms/dependencyagentwindows" -OutFile InstallDependencyAgent-Windows.exe
.\InstallDependencyAgent-Windows.exe /S
```

## Install the Dependency agent on Linux

The Dependency agent is installed on Linux servers from `InstallDependencyAgent-Linux64.bin`, a shell script with a self-extracting binary. You can run the file by using `sh` or add execute permissions to the file itself.

### NOTE

Root access is required to install or configure the agent.

PARAMETER	DESCRIPTION
<code>-help</code>	Get a list of the command-line options.

PARAMETER	DESCRIPTION
-s	Perform a silent installation with no user prompts.
--check	Check permissions and the operating system, but don't install the agent.

For example, to run the installation program with the `-help` parameter, enter **InstallDependencyAgent-Linux64.bin -help**. Install the Linux Dependency agent as root by running the command  
`sh InstallDependencyAgent-Linux64.bin`.

If the Dependency agent fails to start, check the logs for detailed error information. On Linux agents, the log directory is `/var/opt/microsoft/dependency-agent/log`.

Files for the Dependency agent are placed in the following directories:

FILES	LOCATION
Core files	<code>/opt/microsoft/dependency-agent</code>
Log files	<code>/var/opt/microsoft/dependency-agent/log</code>
Config files	<code>/etc/opt/microsoft/dependency-agent/config</code>
Service executable files	<code>/opt/microsoft/dependency-agent/bin/microsoft-dependency-agent</code> <code>/opt/microsoft/dependency-agent/bin/microsoft-dependency-agent-manager</code>
Binary storage files	<code>/var/opt/microsoft/dependency-agent/storage</code>

## Shell script

Use the following sample shell script to download and install the agent:

```
wget --content-disposition https://aka.ms/dependencyagentlinux -O InstallDependencyAgent-Linux64.bin
sudo sh InstallDependencyAgent-Linux64.bin -s
```

## Desired State Configuration

To deploy the Dependency agent using Desired State Configuration (DSC), you can use the `xPSDesiredStateConfiguration` module with the following example code:

```

configuration VMInsights {

    Import-DscResource -ModuleName xPSDesiredStateConfiguration

    $DAPackageLocalPath = "C:\InstallDependencyAgent-Windows.exe"

    Node localhost
    {
        # Download and install the Dependency agent
        xRemoteFile DAPackage
        {
            Uri = "https://aka.ms/dependencyagentwindows"
            DestinationPath = $DAPackageLocalPath
        }

        xPackage DA
        {
            Ensure="Present"
            Name = "Dependency Agent"
            Path = $DAPackageLocalPath
            Arguments = '/S'
            ProductId = ""
            InstalledCheckRegKey =
            "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\DependencyAgent"
            InstalledCheckRegValueName = "DisplayName"
            InstalledCheckRegValueData = "Dependency Agent"
            DependsOn = "[xRemoteFile]DAPackage"
        }
    }
}

```

## Troubleshooting

### VM doesn't appear on the map

If your Dependency agent installation succeeded, but you don't see your computer on the map, diagnose the problem by following these steps.

1. Is the Dependency agent installed successfully? You can validate this by checking to see if the service is installed and running.

**Windows:** Look for the service named "Microsoft Dependency agent."

**Linux:** Look for the running process "microsoft-dependency-agent."

2. Are you on the [Free pricing tier of Log Analytics](#)? The Free plan allows for up to five unique computers. Any subsequent computers won't show up on the map, even if the prior five are no longer sending data.
3. Is the computer sending log and perf data to Azure Monitor Logs? Perform the following query for your computer:

```
Usage | where Computer == "computer-name" | summarize sum(Quantity), any(QuantityUnit) by DataType
```

Did it return one or more results? Is the data recent? If so, your Log Analytics agent is operating correctly and communicating with the service. If not, check the agent on your server: [Log Analytics agent for Windows troubleshooting](#) or [Log Analytics agent for Linux troubleshooting](#).

### Computer appears on the map but has no processes

If you see your server on the map, but it has no process or connection data, that indicates that the Dependency agent is installed and running, but the kernel driver didn't load.

Check the C:\Program Files\Microsoft Dependency Agent\logs\wrapper.log file (Windows) or /var/opt/microsoft/dependency-agent/log/service.log file (Linux). The last lines of the file should indicate why the kernel didn't load. For example, the kernel might not be supported on Linux if you updated your kernel.

## Next steps

Now that monitoring is enabled for your virtual machines, this information is available for analysis with VM insights.

- To view discovered application dependencies, see [View VM insights Map](#).
- To identify bottlenecks and overall utilization with your VM's performance, see [View Azure VM performance](#).

# Use the Map feature of VM insights to understand application components

9/21/2022 • 7 minutes to read • [Edit Online](#)

In VM insights, you can view discovered application components on Windows and Linux virtual machines (VMs) that run in Azure or your environment. You can observe the VMs in two ways. View a map directly from a VM or view a map from Azure Monitor to see the components across groups of VMs. This article will help you understand these two viewing methods and how to use the Map feature.

For information about configuring VM insights, see [Enable VM insights](#).

## Prerequisites

To enable the map feature in VM insights, the virtual machine requires one of the following. See [Enable VM insights on unmonitored machine](#) for details on each.

- Azure Monitor agent with **processes and dependencies** enabled.
- Log Analytics agent enabled for VM insights.

### WARNING

Collecting duplicate data from a single machine with both the Azure Monitor agent and Log Analytics agent can result in the map feature of VM insights being inaccurate since it does not check for duplicate data.

See [Migrate from Log Analytics agent](#) for more information.

## Introduction to the Map experience

Before diving into the Map experience, you should understand how it presents and visualizes information. Whether you select the Map feature directly from a VM or from Azure Monitor, the Map feature presents a consistent experience. The only difference is that from Azure Monitor, one map shows all the members of a multiple-tier application or cluster.

The Map feature visualizes the VM dependencies by discovering running processes that have:

- Active network connections between servers.
- Inbound and outbound connection latency.
- Ports across any TCP-connected architecture over a specified time range.

Expand a VM to show process details and only those processes that communicate with the VM. The client group shows the count of front-end clients that connect into the VM. The server-port groups show the count of back-end servers the VM connects to. Expand a server-port group to see the detailed list of servers that connect over that port.

When you select the VM, the **Properties** pane on the right shows the VM's properties. Properties include system information reported by the operating system, properties of the Azure VM, and a doughnut chart that summarizes the discovered connections.

» DC01.Corp.Swanson.Local  
Machine Summary

Fully Qualified Domain Name  
DC01.Corp.Swanson.Local

Operating System  
Windows Server 2016 Version 1607 build 14393

Ipv4 Addresses  
10.99.99.1/24

> More Properties

▼ Machine Dependencies

Connected Servers  
31  
Connected Client  
0

> Azure VM Properties

Properties Log Events Alerts

On the right side of the pane, select **Log Events** to show a list of data that the VM has sent to Azure Monitor. This data is available for querying. Select any record type to open the **Logs** page, where you see the results for that record type. You also see a preconfigured query that's filtered against the VM.

» DC01.Corp.Swanson.Local  
Machine Log Events

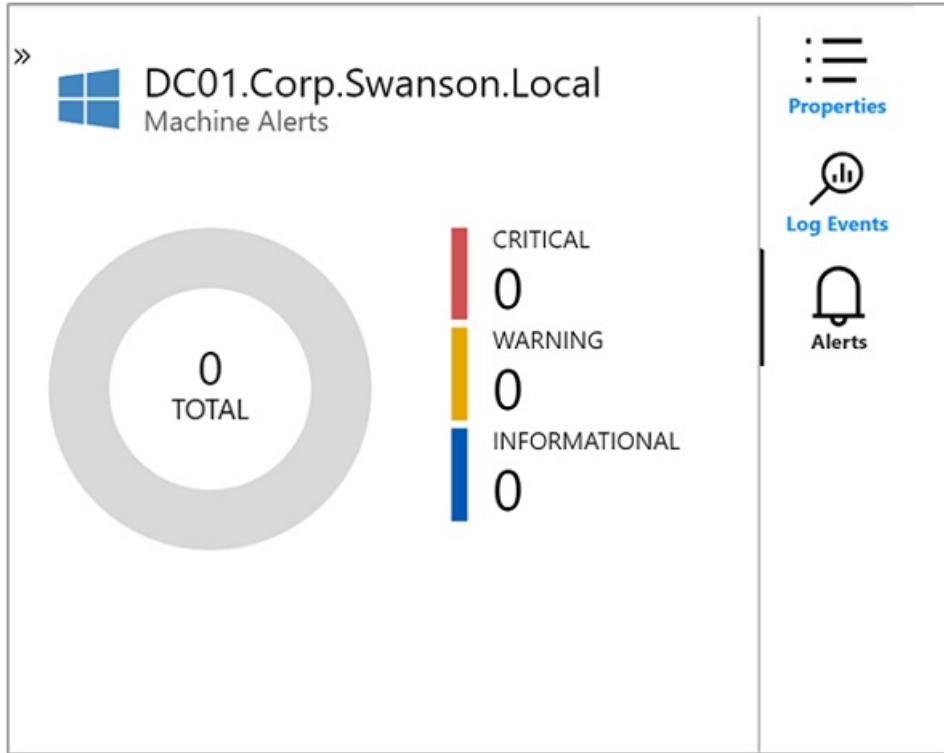
Select an event type to open in Log Analytics

EVENT TYPE	COUNT
Alert	5
Heartbeat	29
Perf	7334
ServiceMapComputer_CL	5
ServiceMapProcess_CL	9
Usage	13
VMConnection	556

Properties Log Events Alerts

Close the **Logs** page and return to the **Properties** pane. There, select **Alerts** to view VM health-criteria alerts. The Map feature integrates with Azure Alerts to show alerts for the selected server in the selected time range.

The server displays an icon for current alerts, and the **Machine Alerts** pane lists the alerts.



To make the Map feature display relevant alerts, create an alert rule that applies to a specific computer:

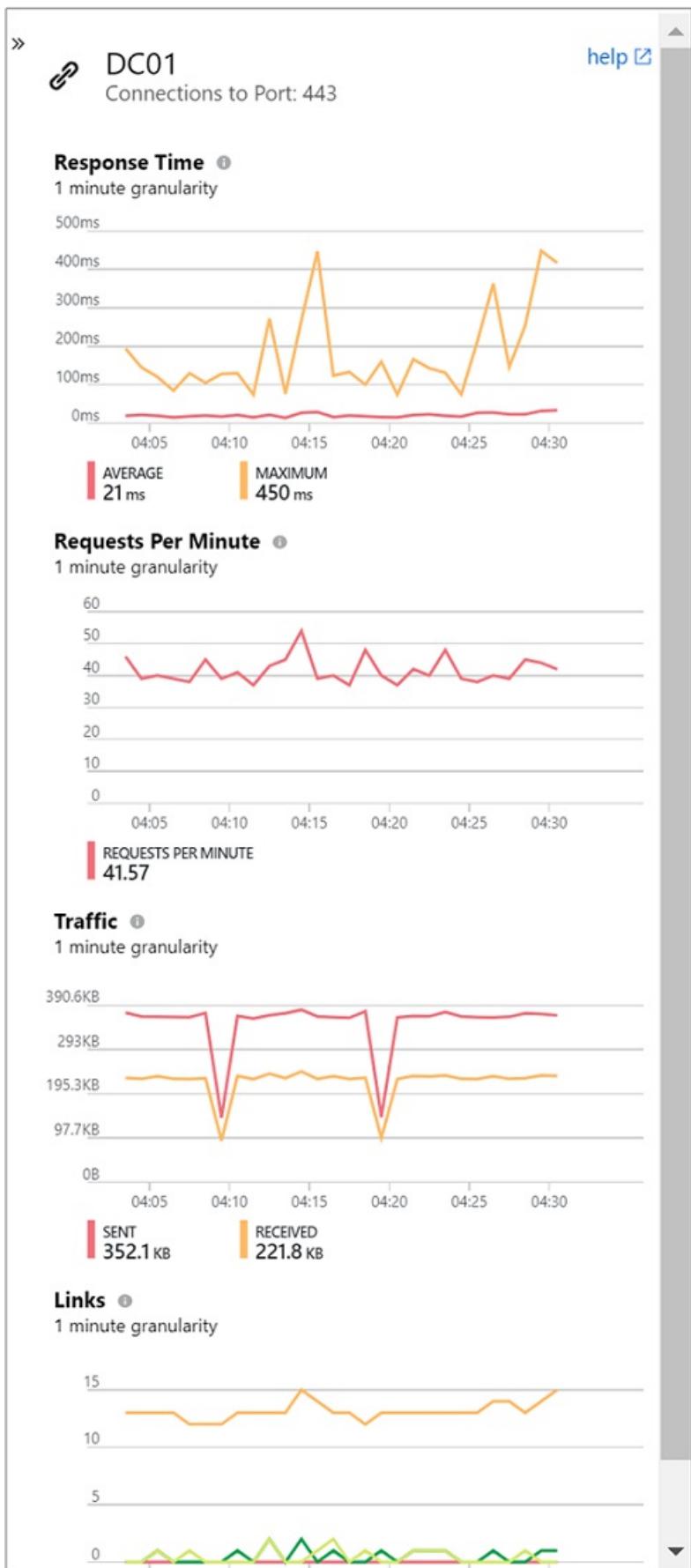
- Include a clause to group alerts by computer (for example, by **Computer interval 1 minute**).
- Base the alert on a metric.

For more information about Azure Alerts and creating alert rules, see [Unified alerts in Azure Monitor](#).

In the upper-right corner, the **Legend** option describes the symbols and roles on the map. For a closer look at your map and to move it around, use the zoom controls in the lower-right corner. You can set the zoom level and fit the map to the size of the page.

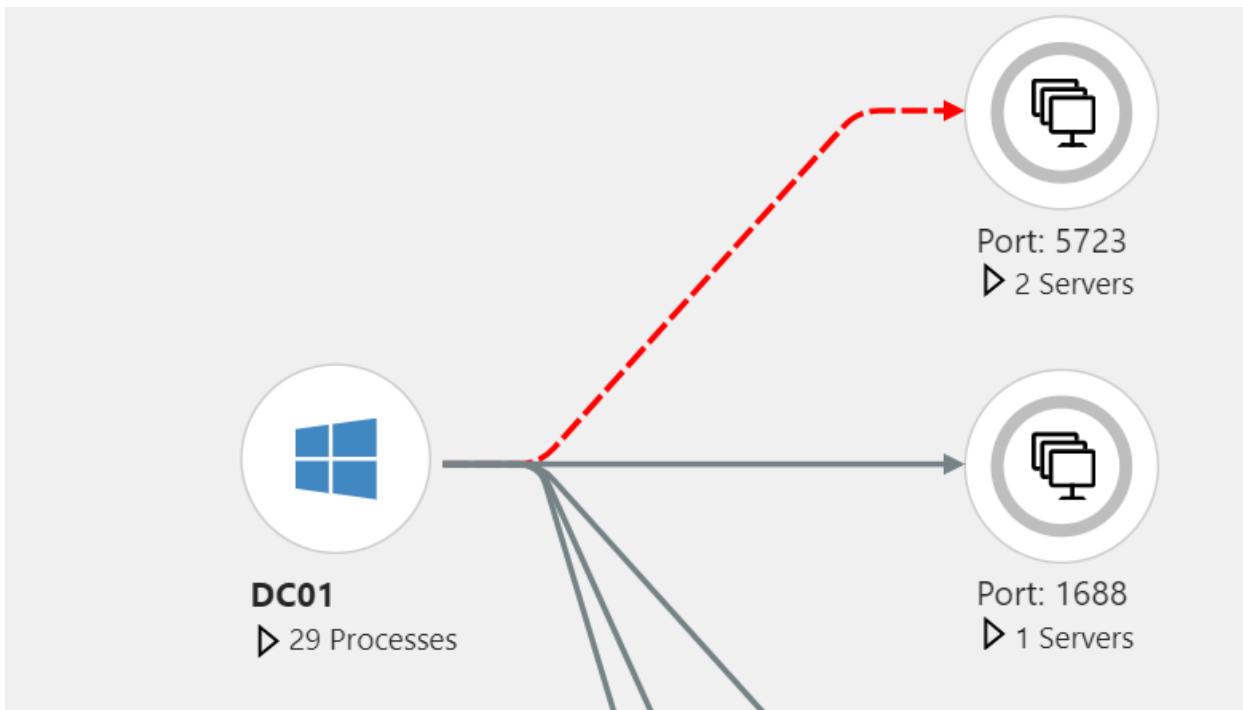
## Connection metrics

The **Connections** pane displays standard metrics for the selected connection from the VM over the TCP port. The metrics include response time, requests per minute, traffic throughput, and links.



## Failed connections

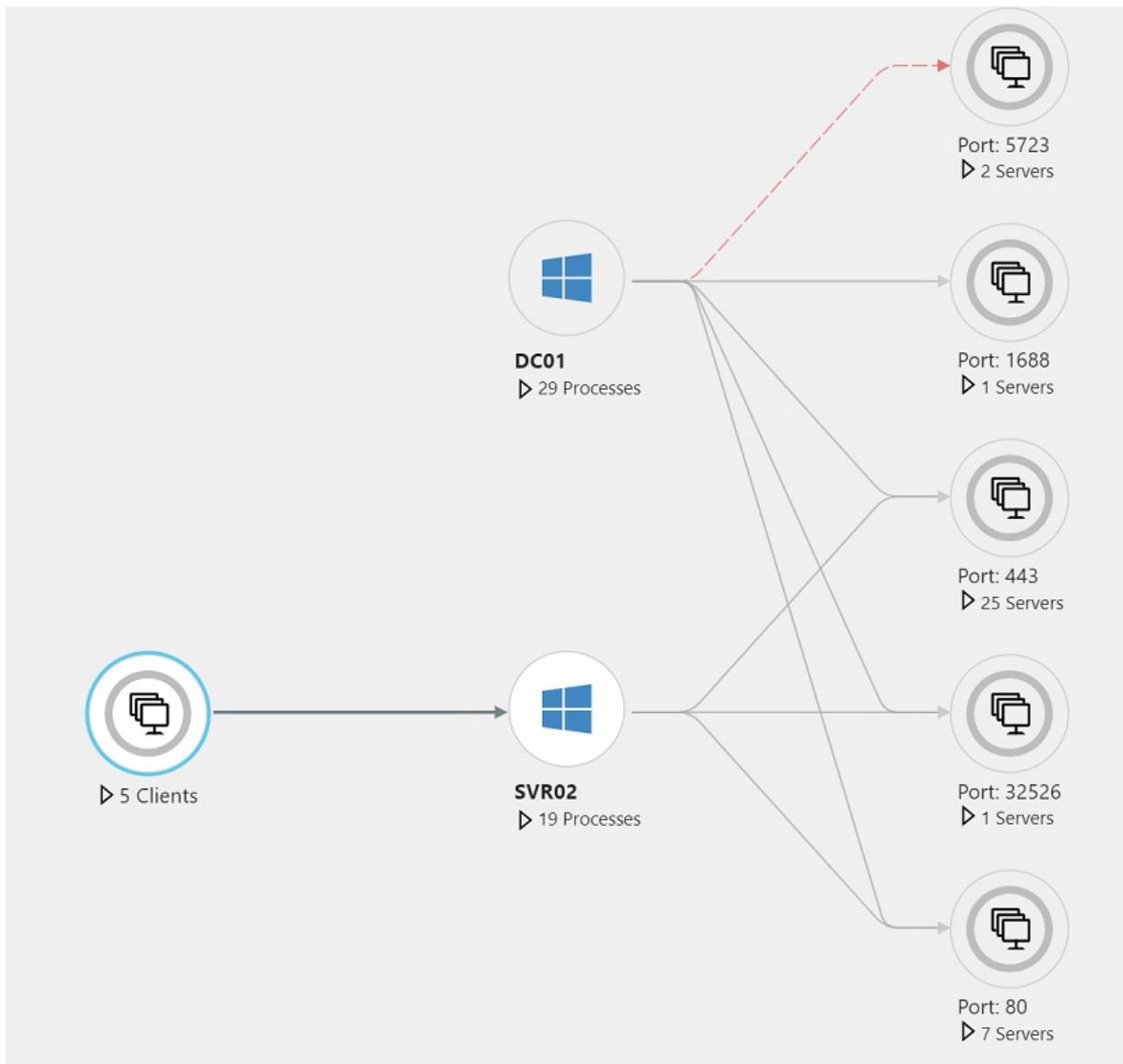
The map shows failed connections for processes and computers. A dashed red line indicates a client system is failing to reach a process or port. For systems that use the Dependency agent, the agent reports on failed connection attempts. The Map feature monitors a process by observing TCP sockets that fail to establish a connection. This failure could result from a firewall, a misconfiguration in the client or server, or an unavailable remote service.



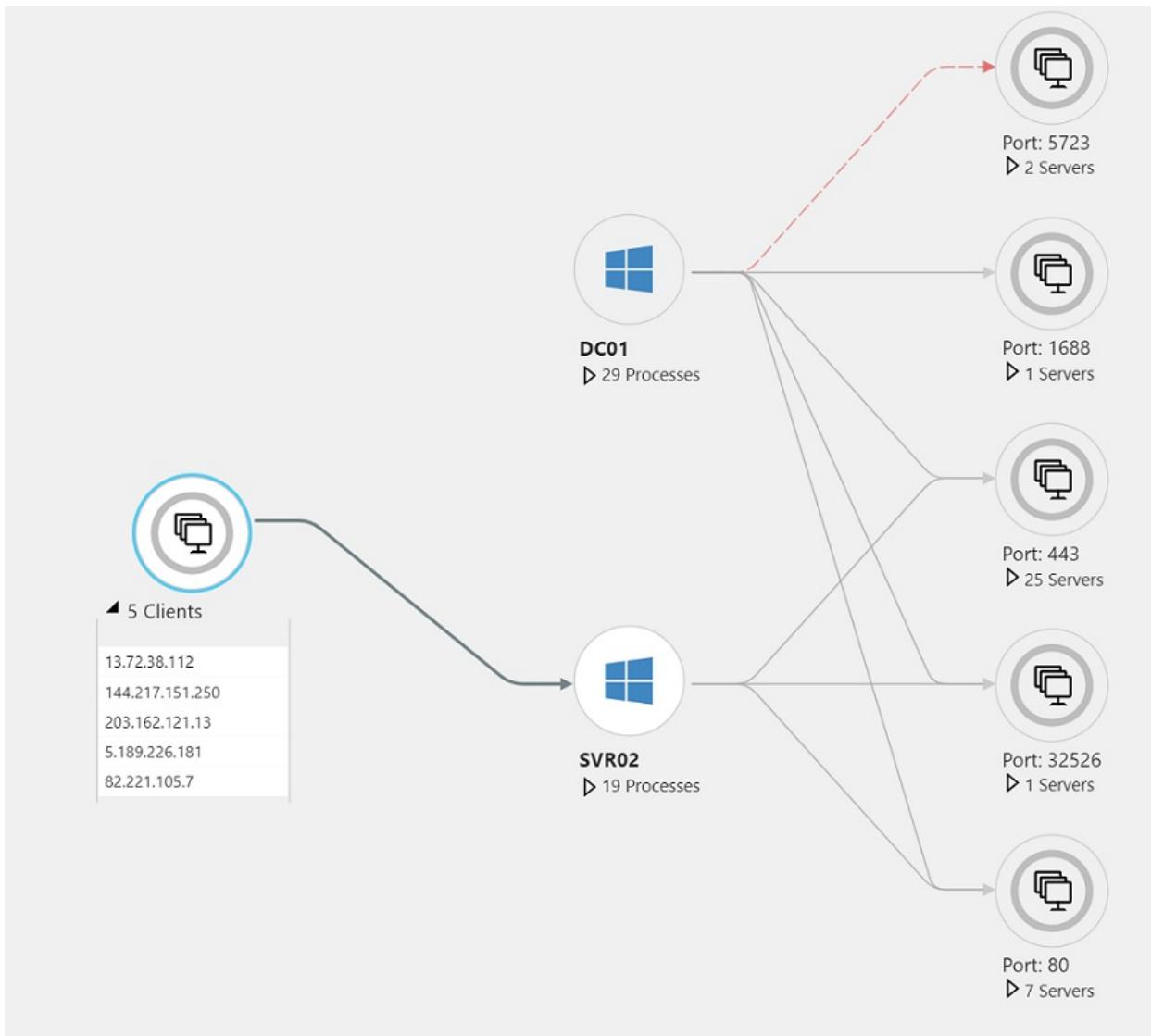
Understanding failed connections can help you troubleshoot, validate migration, analyze security, and understand the overall architecture of the service. Failed connections are sometimes harmless, but they often point to a problem. Connections might fail, for example, when a failover environment suddenly becomes unreachable or when two application tiers can't communicate with each other after a cloud migration.

### **Client groups**

On the map, client groups represent client machines that connect to the mapped machine. A single client group represents the clients for an individual process or machine.



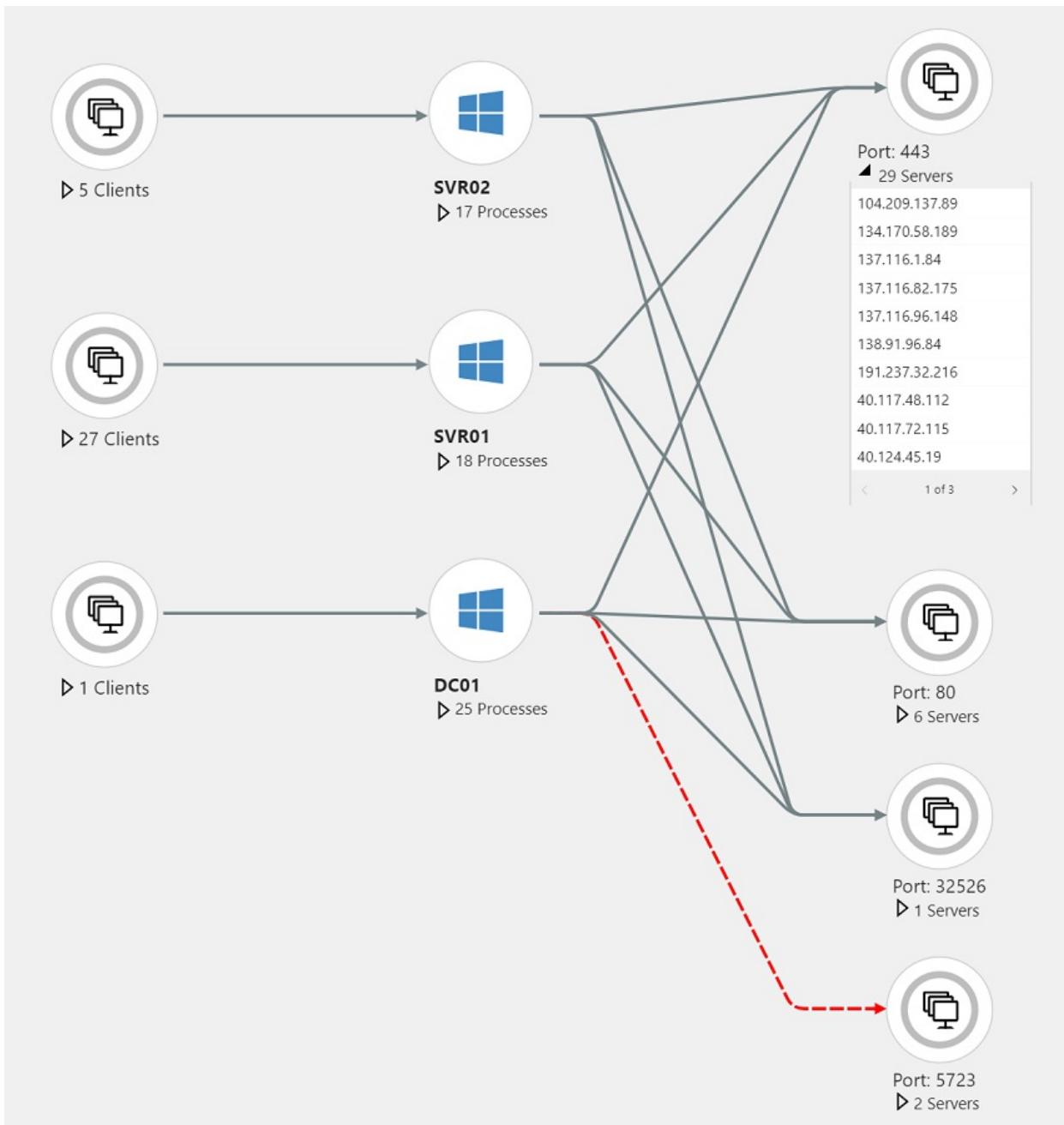
To see the monitored clients and IP addresses of the systems in a client group, select the group. The contents of the group appear below.



If the group includes monitored and unmonitored clients, you can select the appropriate section of the group's doughnut chart to filter the clients.

### Server-port groups

Server-port groups represent ports on servers that have inbound connections from the mapped machine. The group contains the server port and a count of the number of servers that have connections to that port. Select the group to see the individual servers and connections.



If the group includes monitored and unmonitored servers, you can select the appropriate section of the group's doughnut chart to filter the servers.

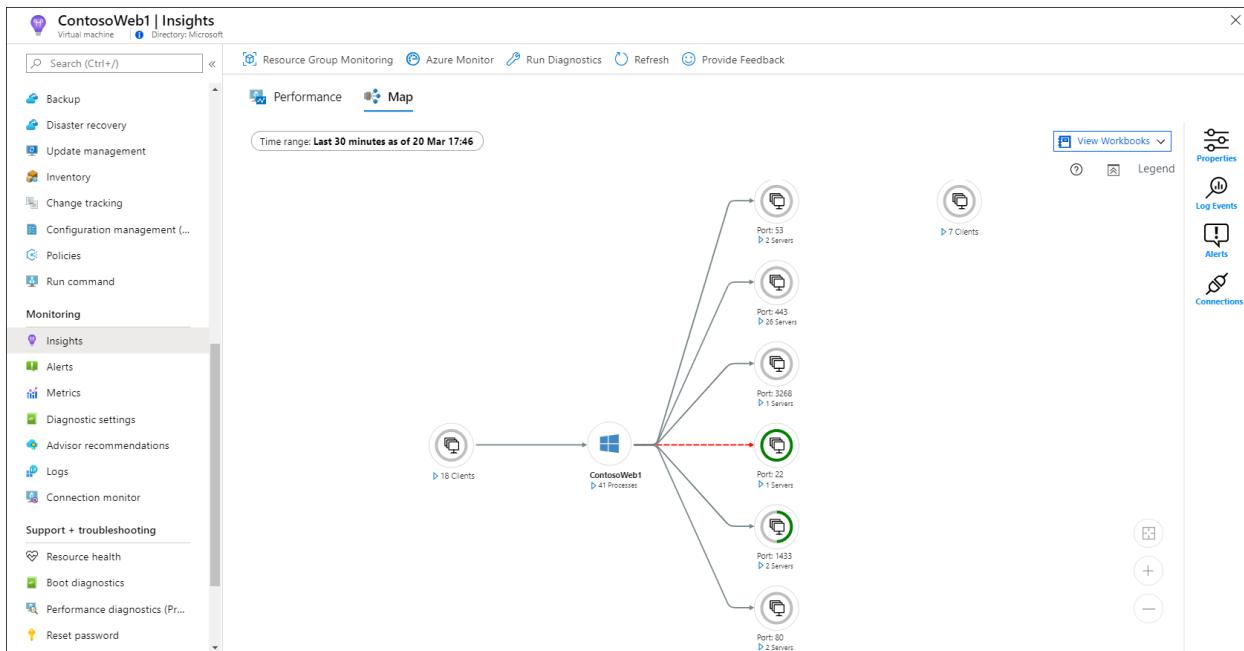
## View a map from a VM

To access VM insights directly from a VM:

1. In the Azure portal, select **Virtual Machines**.
2. From the list, choose a VM. In the **Monitoring** section, choose **Insights**.
3. Select the **Map** tab.

The map visualizes the VM's dependencies by discovering running process groups and processes that have active network connections over a specified time range.

By default, the map shows the last 30 minutes. If you want to see how dependencies looked in the past, you can query for historical time ranges of up to one hour. To run the query, use the **TimeRange** selector in the upper-left corner. You might run a query, for example, during an incident or to see the status before a change.



## View a map from a virtual machine scale set

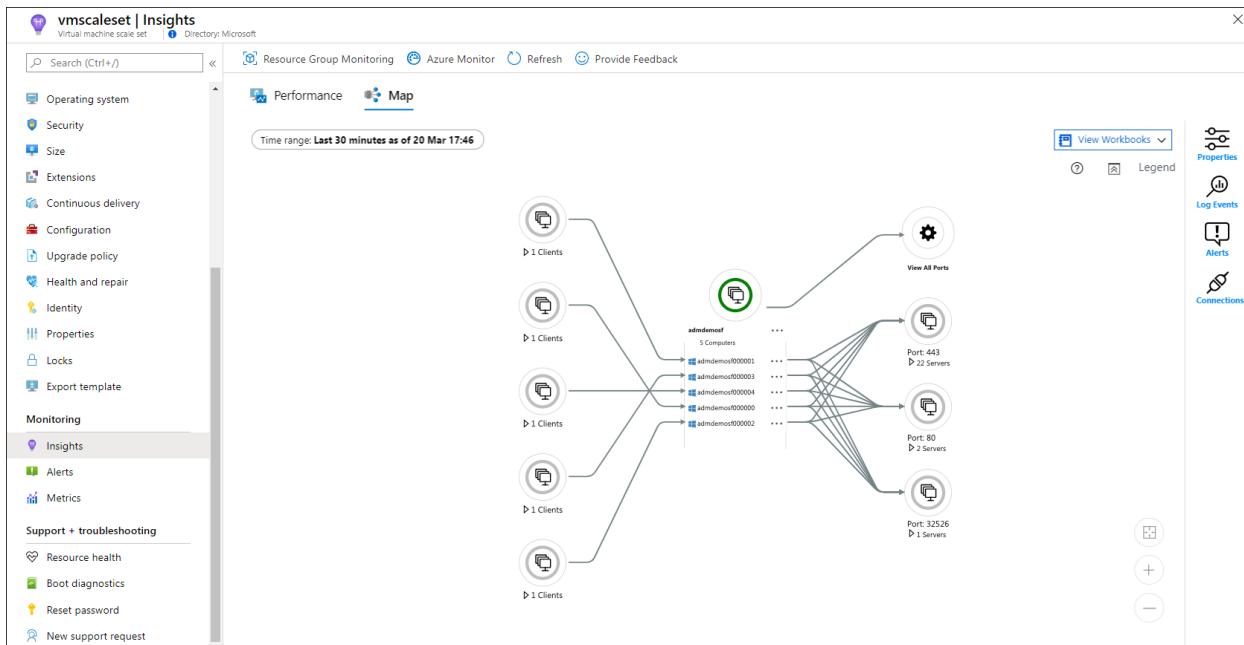
To access VM insights directly from a virtual machine scale set:

1. In the Azure portal, select **Virtual machine scale sets**.
2. From the list, choose a VM. Then in the **Monitoring** section, choose **Insights**.
3. Select the **Map** tab.

The map visualizes all instances in the scale set as a group node along with the group's dependencies. The expanded node lists the instances in the scale set. You can scroll through these instances 10 at a time.

To load a map for a specific instance, first select that instance on the map. Then select the **ellipsis** button (...) to the right and choose **Load Server Map**. In the map that appears, you see process groups and processes that have active network connections over a specified time range.

By default, the map shows the last 30 minutes. If you want to see how dependencies looked in the past, you can query for historical time ranges of up to one hour. To run the query, use the **TimeRange** selector. You might run a query, for example, during an incident or to see the status before a change.



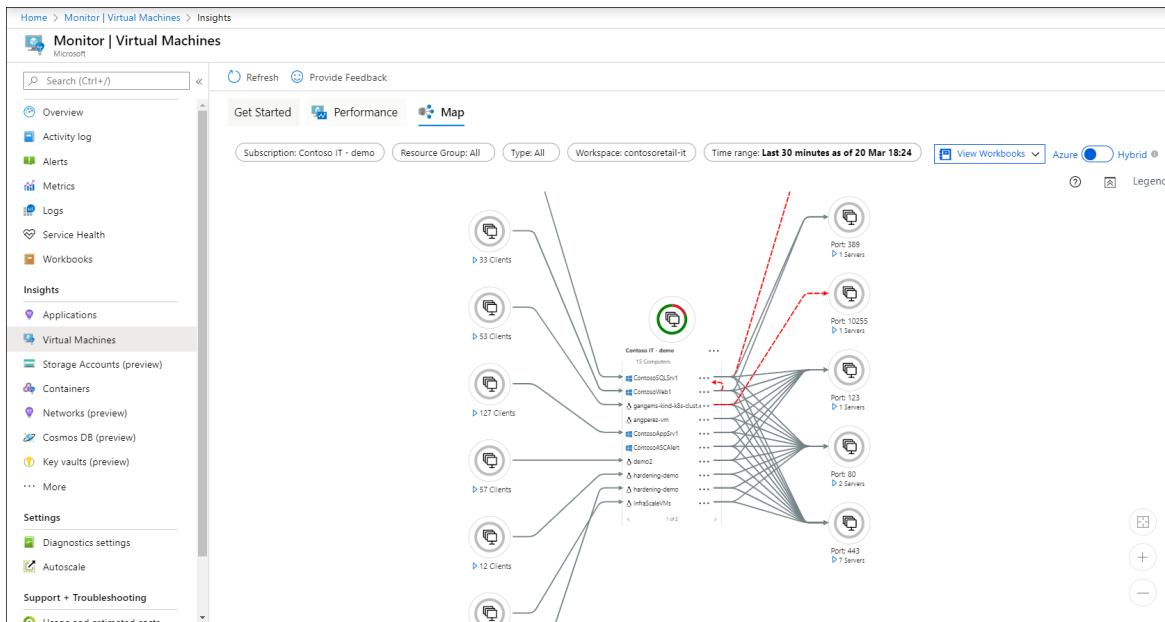
## NOTE

You can also access a map for a specific instance from the **Instances** view for your virtual machine scale set. In the **Settings** section, go to **Instances > Insights**.

## View a map from Azure Monitor

In Azure Monitor, the Map feature provides a global view of your VMs and their dependencies. To access the Map feature in Azure Monitor:

1. In the Azure portal, select **Monitor**.
2. In the **Insights** section, choose **Virtual Machines**.
3. Select the **Map** tab.



Choose a workspace by using the **Workspace** selector at the top of the page. If you have more than one Log Analytics workspace, choose the workspace that's enabled with the solution and that has VMs reporting to it.

The **Group** selector returns subscriptions, resource groups, **computer groups**, and virtual machine scale sets of computers that are related to the selected workspace. Your selection applies only to the Map feature and doesn't carry over to Performance or Health.

By default, the map shows the last 30 minutes. If you want to see how dependencies looked in the past, you can query for historical time ranges of up to one hour. To run the query, use the **TimeRange** selector. You might run a query, for example, during an incident or to see the status before a change.

## Next steps

To identify bottlenecks, check performance, and understand overall utilization of your VMs, see [View performance status for VM insights](#).

# How to chart performance with VM insights

9/21/2022 • 6 minutes to read • [Edit Online](#)

VM insights includes a set of performance charts that target several key performance indicators (KPIs) to help you determine how well a virtual machine is performing. The charts show resource utilization over a period of time so you can identify bottlenecks, anomalies, or switch to a perspective listing each machine to view resource utilization based on the metric selected. While there are numerous elements to consider when dealing with performance, VM insights monitors key operating system performance indicators related to processor, memory, network adapter, and disk utilization. Performance complements the health monitoring feature and helps expose issues that indicate a possible system component failure, support tuning and optimization to achieve efficiency, or support capacity planning.

## Limitations

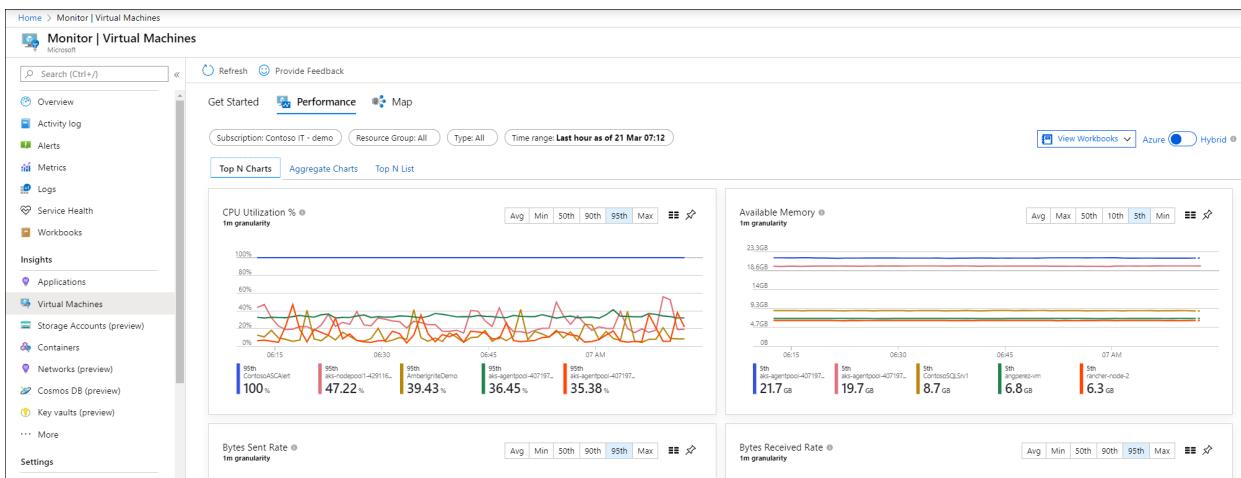
Following are limitations in performance collection with VM insights.

- **Available memory** is not available for virtual machines running Red Hat Linux (RHEL) 6. This metric is calculated from **MemAvailable** which was introduced in [kernel version 3.14](#).
- Metrics are only available for data disks on Linux virtual machines using XFS filesystem or EXT filesystem family (EXT2, EXT3, EXT4).

## Multi-VM perspective from Azure Monitor

From Azure Monitor, the Performance feature provides a view of all monitored VMs deployed across workgroups in your subscriptions or in your environment. To access from Azure Monitor, perform the following steps.

1. In the Azure portal, select **Monitor**.
2. Choose **Virtual Machines** in the **Solutions** section.
3. Select the **Performance** tab.



On the **Top N Charts** tab, if you have more than one Log Analytics workspace, choose the workspace enabled with the solution from the **Workspace** selector at the top of the page. The **Group** selector will return subscriptions, resource groups, [computer groups](#), and virtual machine scale sets of computers related to the selected workspace that you can use to further filter results presented in the charts on this page and across the other pages. Your selection only applies to the Performance feature and does not carry over to Health or Map.

By default, the charts show the last 24 hours. Using the **TimeRange** selector, you can query for historical time ranges of up to 30 days to show how performance looked in the past.

The five capacity utilization charts shown on the page are:

- CPU Utilization % - shows the top five machines with the highest average processor utilization
- Available Memory - shows the top five machines with the lowest average amount of available memory
- Logical Disk Space Used % - shows the top five machines with the highest average disk space used % across all disk volumes
- Bytes Sent Rate - shows the top five machines with highest average of bytes sent
- Bytes Receive Rate - shows the top five machines with highest average of bytes received

Clicking on the pin icon at the upper right-hand corner of any one of the five charts will pin the selected chart to the last Azure dashboard you last viewed. From the dashboard, you can resize and reposition the chart.

Selecting the chart from the dashboard will redirect you to VM insights and load the correct scope and view.

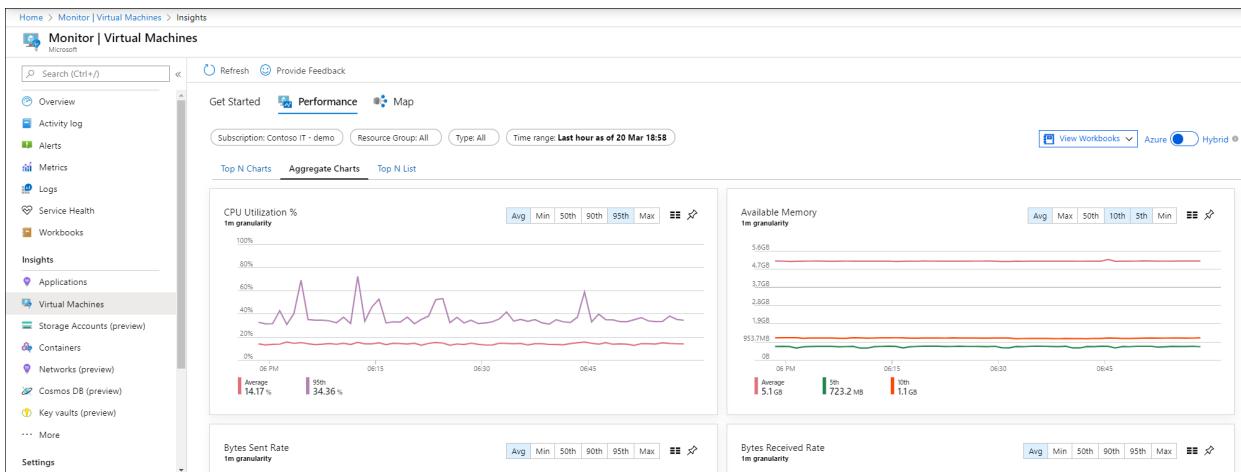
Clicking on the icon located to the left of the pin icon on any one of the five charts opens the **Top N List** view. Here you see the resource utilization for that performance metric by individual VM in a list view and which machine is trending highest.

The screenshot shows the Azure Monitor interface for Virtual Machines. The left sidebar has sections for Overview, Activity log, Alerts, Metrics, Logs, Service Health, Workbooks, and Insights. Under Insights, Virtual Machines is selected. The main area shows a Performance chart with various metrics like CPU Utilization %, Memory Usage, and Disk I/O. Below the chart is a search bar and a dropdown for Metric (set to CPU Utilization %). A table lists the top 10 VMs based on CPU Utilization, showing their names, average utilization, and 5th, 10th, 90th, and 95th percentile utilization. Each row includes a small icon and a bar chart representing the trend. The table shows 32 items in total. The right side of the interface has a Properties pane expanded, showing details for the selected VM.

NAME	AVER...	5TH	10TH	90TH	95TH	TREND 95TH (1 BAR...)	TYPE
ContosoASCAAlert	100%	100%	100%	100%	100%		Virtual machine
aks-nodepool1-42911611-2	29%	16%	17%	47%	53%		Virtual machine
aks-agentpool-40719753-2	33%	31%	31%	35%	35%		Virtual machine
aks-agentpool-40719753-1	14%	6%	6%	28%	33%		Virtual machine
ContosoSQLSrv1	7%	3%	3%	11%	30%		Virtual machine
AmberIgniteDemo	12%	5%	6%	25%	28%		Virtual machine
node-4	14%	9%	9%	25%	27%		Virtual machine
ContosoWeb1	20%	18%	18%	23%	24%		Virtual machine

When you click on the virtual machine, the **Properties** pane is expanded on the right to show the properties of the item selected, such as system information reported by the operating system, properties of the Azure VM, etc. Clicking on one of the options under the **Quick Links** section will redirect you to that feature directly from the selected VM.

Switch to the Aggregated Charts tab to view the performance metrics filtered by average or percentiles measures.



The following capacity utilization charts are provided:

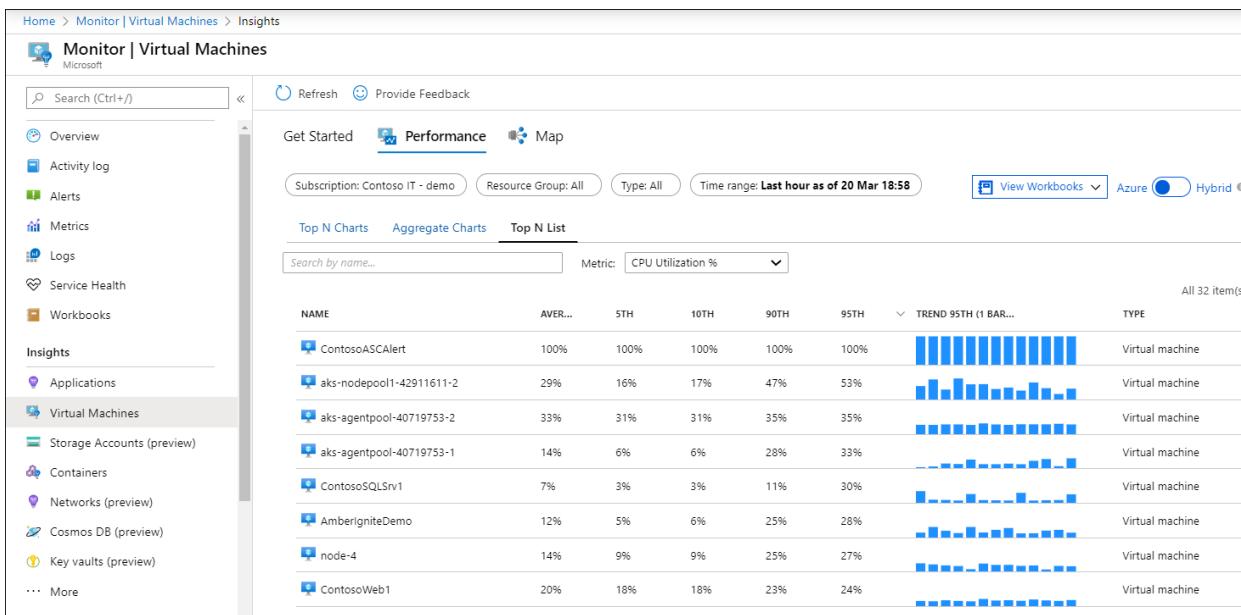
- CPU Utilization % - defaults showing the average and top 95th percentile
- Available Memory - defaults showing the average, top 5th, and 10th percentile
- Logical Disk Space Used % - defaults showing the average and 95th percentile
- Bytes Sent Rate - defaults showing average bytes sent
- Bytes Receive Rate - defaults showing average bytes received

You can also change the granularity of the charts within the time range by selecting **Avg**, **Min**, **Max**, **50th**, **90th**, and **95th** in the percentile selector.

To view the resource utilization by individual VM in a list view and see which machine is trending with highest utilization, select the **Top N List** tab. The **Top N List** page shows the top 20 machines sorted by the most utilized by 95th percentile for the metric *CPU Utilization %*. You can see more machines by selecting **Load More**, and the results expand to show the top 500 machines.

## NOTE

The list cannot show more than 500 machines at a time.



The screenshot shows the Azure Monitor Virtual Machines Insights page. On the left, there's a navigation sidebar with links like Overview, Activity log, Alerts, Metrics, Logs, Service Health, Workbooks, Insights, Applications, Virtual Machines (which is selected), Storage Accounts (preview), Containers, Networks (preview), Cosmos DB (preview), Key vaults (preview), and More. The main area has tabs for Get Started, Performance (which is selected), and Map. It shows a search bar, filter options for Subscription, Resource Group, Type, and Time range (Last hour as of 20 Mar 18:58), and buttons for View Workbooks, Azure, and Hybrid. Below these are three tabs: Top N Charts, Aggregate Charts, and Top N List (which is selected). A search bar under the tabs allows filtering by name. A dropdown menu for Metric shows CPU Utilization % as the current selection. The main content area displays a table of 32 virtual machines with columns for Name, Average, 5th, 10th, 90th, 95th, Trend 95th (1 Bar...), and Type. Each row includes a small icon of the VM and its name. The 'Trend 95th (1 Bar...)' column shows blue bars representing the 95th percentile trend over time. The 'Type' column indicates all listed VMs are Virtual machines.

To filter the results on a specific virtual machine in the list, enter its computer name in the **Search by name** textbox.

If you would rather view utilization from a different performance metric, from the **Metric** drop-down list select **Available Memory**, **Logical Disk Space Used %**, **Network Received Byte/s**, or **Network Sent Byte/s** and the list updates to show utilization scoped to that metric.

Selecting a virtual machine from the list opens the **Properties** panel on the right-side of the page and from here you can select **Performance detail**. The **Virtual Machine Detail** page opens and is scoped to that VM, similar in experience when accessing VM Insights Performance directly from the Azure VM.

## View performance directly from an Azure VM

To access directly from a virtual machine, perform the following steps.

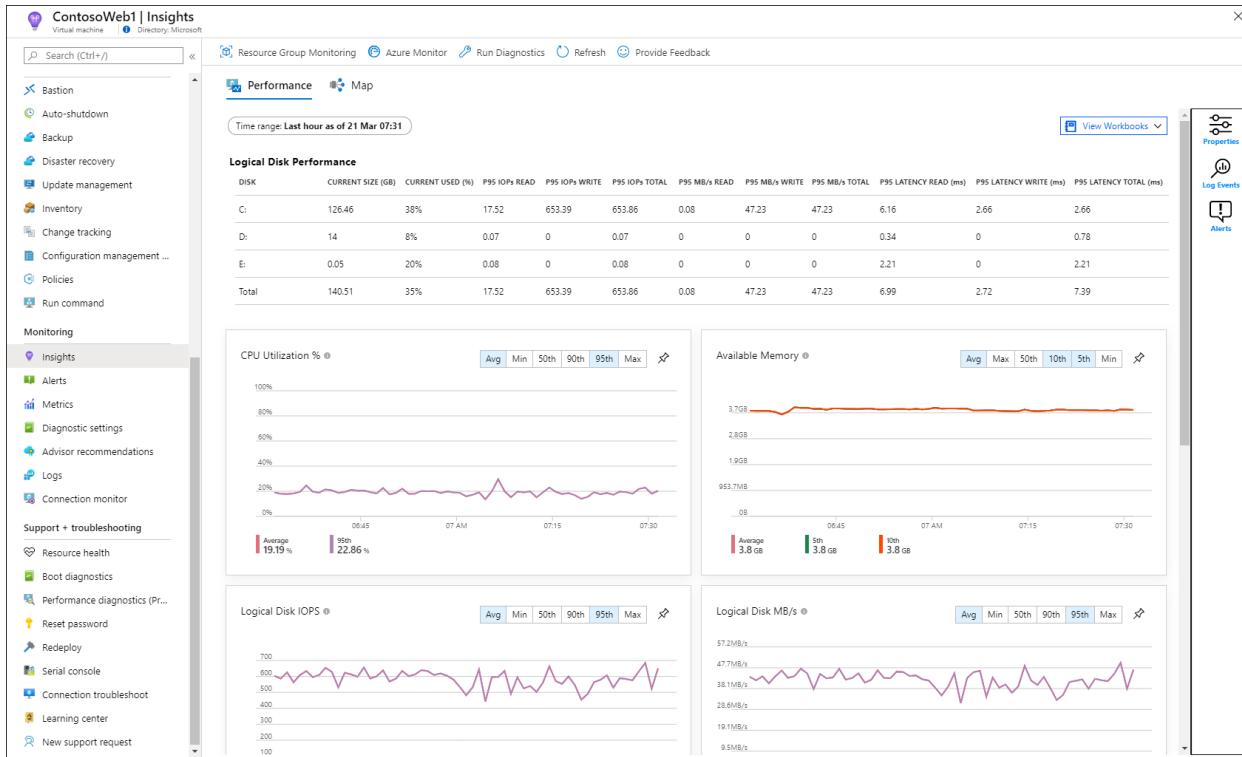
1. In the Azure portal, select **Virtual Machines**.
2. From the list, choose a VM and in the **Monitoring** section choose **Insights**.
3. Select the **Performance** tab.

This page not only includes performance utilization charts, but also a table showing for each logical disk discovered, its capacity, utilization, and total average by each measure.

The following capacity utilization charts are provided:

- CPU Utilization % - defaults showing the average and top 95th percentile
- Available Memory - defaults showing the average, top 5th, and 10th percentile
- Logical Disk Space Used % - defaults showing the average and 95th percentile
- Logical Disk IOPS - defaults showing the average and 95th percentile
- Logical Disk MB/s - defaults showing the average and 95th percentile
- Max Logical Disk Used % - defaults showing the average and 95th percentile
- Bytes Sent Rate - defaults showing average bytes sent
- Bytes Receive Rate - defaults showing average bytes received

Clicking on the pin icon at the upper right-hand corner of any one of the charts pins the selected chart to the last Azure dashboard you viewed. From the dashboard, you can resize and reposition the chart. Selecting the chart from the dashboard redirects you to VM insights and loads the performance detail view for the VM.



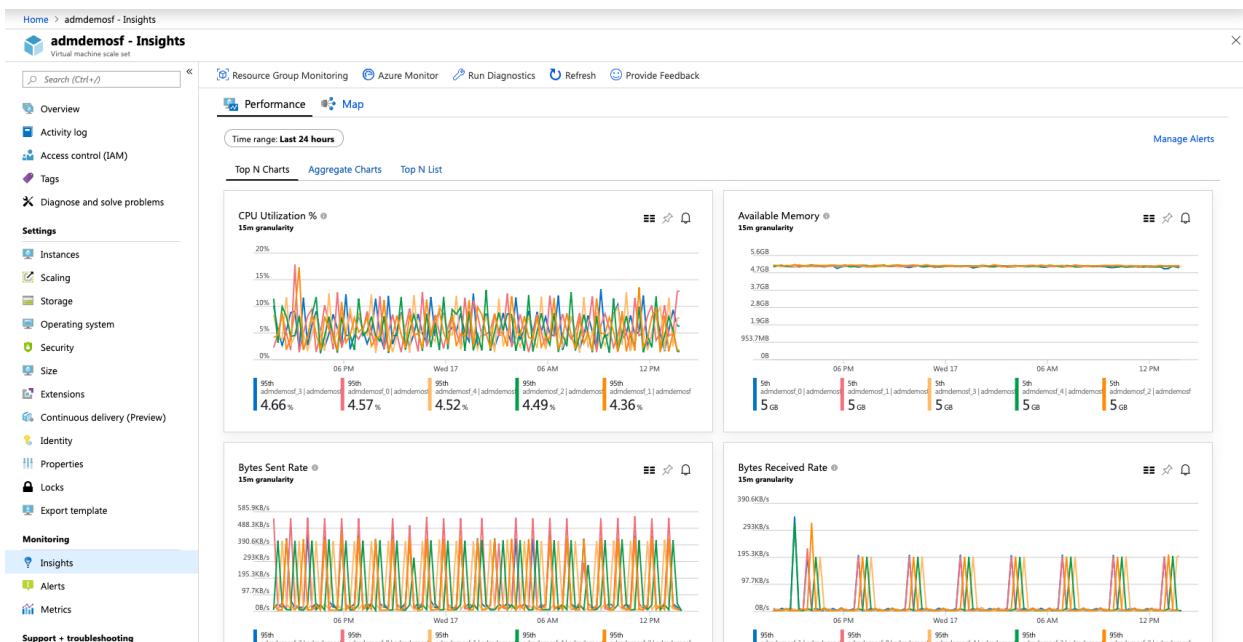
## View performance directly from an Azure virtual machine scale set

To access directly from an Azure virtual machine scale set, perform the following steps.

1. In the Azure portal, select **Virtual machine scale sets**.
2. From the list, choose a VM and in the **Monitoring** section choose **Insights** to view the **Performance** tab.

This page loads the Azure Monitor performance view, scoped to the selected scale set. This enables you to see the Top N Instances in the scale set across the set of monitored metrics, view the aggregate performance across the scale set, and see the trends for selected metrics across the individual instances in the scale set. Selecting an instance from the list view lets you load its map or navigate into a detailed performance view for that instance.

Clicking on the pin icon at the upper right-hand corner of any one of the charts pins the selected chart to the last Azure dashboard you viewed. From the dashboard, you can resize and reposition the chart. Selecting the chart from the dashboard redirects you to VM insights and loads the performance detail view for the VM.



### NOTE

You can also access a detailed performance view for a specific instance from the Instances view for your scale set. Navigate to **Instances** under the **Settings** section, and then choose **Insights**.

## Next steps

- Learn how to use [Workbooks](#) that are included with VM insights to further analyze performance and network metrics.
- To learn about discovered application dependencies, see [View VM insights Map](#).

# How to query logs from VM insights

9/21/2022 • 18 minutes to read • [Edit Online](#)

VM insights collects performance and connection metrics, computer and process inventory data, and health state information and forwards it to the Log Analytics workspace in Azure Monitor. This data is available for [query](#) in Azure Monitor. You can apply this data to scenarios that include migration planning, capacity analysis, discovery, and on-demand performance troubleshooting.

## Map records

One record is generated per hour for each unique computer and process, in addition to the records that are generated when a process or computer starts or is added to VM insights. The fields and values in the ServiceMapComputer\_CL events map to fields of the Machine resource in the ServiceMap Azure Resource Manager API. The fields and values in the ServiceMapProcess\_CL events map to the fields of the Process resource in the ServiceMap Azure Resource Manager API. The ResourceName\_s field matches the name field in the corresponding Resource Manager resource.

There are internally generated properties you can use to identify unique processes and computers:

- Computer: Use *ResourceId* or *ResourceName\_s* to uniquely identify a computer within a Log Analytics workspace.
- Process: Use *ResourceId* to uniquely identify a process within a Log Analytics workspace. *ResourceName\_s* is unique within the context of the machine on which the process is running (*MachineResourceName\_s*)

Because multiple records can exist for a specified process and computer in a specified time range, queries can return more than one record for the same computer or process. To include only the most recent record, add

```
| summarize arg_max(TimeGenerated, *) by ResourceId
```

## Connections and ports

The Connection Metrics feature introduces two new tables in Azure Monitor logs - VMConnection and VMBoundPort. These tables provide information about the connections for a machine (inbound and outbound), as well as the server ports that are open/active on them. ConnectionMetrics are also exposed via APIs that provide the means to obtain a specific metric during a time window. TCP connections resulting from *accepting* on a listening socket are inbound, while those created by *connecting* to a given IP and port are outbound. The direction of a connection is represented by the Direction property, which can be set to either **inbound** or **outbound**.

Records in these tables are generated from data reported by the Dependency Agent. Every record represents an observation over a 1-minute time interval. The TimeGenerated property indicates the start of the time interval. Each record contains information to identify the respective entity, that is, connection or port, as well as metrics associated with that entity. Currently, only network activity that occurs using TCP over IPv4 is reported.

### Common fields and conventions

The following fields and conventions apply to both VMConnection and VMBoundPort:

- Computer: Fully-qualified domain name of reporting machine
- AgentId: The unique identifier for a machine with the Log Analytics agent
- Machine: Name of the Azure Resource Manager resource for the machine exposed by ServiceMap. It is of the form *m-{GUID}*, where *GUID* is the same GUID as AgentId
- Process: Name of the Azure Resource Manager resource for the process exposed by ServiceMap. It is of the form *p-{hex string}*. Process is unique within a machine scope and to generate a unique process ID across

machines, combine Machine and Process fields.

- ProcessName: Executable name of the reporting process.
- All IP addresses are strings in IPv4 canonical format, for example `13.107.3.160`

To manage cost and complexity, connection records do not represent individual physical network connections. Multiple physical network connections are grouped into a logical connection, which is then reflected in the respective table. Meaning, records in `VMConnection` table represent a logical grouping and not the individual physical connections that are being observed. Physical network connection sharing the same value for the following attributes during a given one-minute interval, are aggregated into a single logical record in `VMConnection`.

PROPERTY	DESCRIPTION
Direction	Direction of the connection, value is <i>inbound</i> or <i>outbound</i>
Machine	The computer FQDN
Process	Identity of process or groups of processes, initiating/accepting the connection
Sourcelp	IP address of the source
Destinationlp	IP address of the destination
DestinationPort	Port number of the destination
Protocol	Protocol used for the connection. Values is <i>tcp</i> .

To account for the impact of grouping, information about the number of grouped physical connections is provided in the following properties of the record:

PROPERTY	DESCRIPTION
LinksEstablished	The number of physical network connections that have been established during the reporting time window
LinksTerminated	The number of physical network connections that have been terminated during the reporting time window
LinksFailed	The number of physical network connections that have failed during the reporting time window. This information is currently available only for outbound connections.
LinksLive	The number of physical network connections that were open at the end of the reporting time window

## Metrics

In addition to connection count metrics, information about the volume of data sent and received on a given logical connection or network port are also included in the following properties of the record:

PROPERTY	DESCRIPTION
BytesSent	Total number of bytes that have been sent during the reporting time window

PROPERTY	DESCRIPTION
BytesReceived	Total number of bytes that have been received during the reporting time window
Responses	The number of responses observed during the reporting time window.
ResponseTimeMax	The largest response time (milliseconds) observed during the reporting time window. If no value, the property is blank.
ResponseTimeMin	The smallest response time (milliseconds) observed during the reporting time window. If no value, the property is blank.
ResponseTimeSum	The sum of all response times (milliseconds) observed during the reporting time window. If no value, the property is blank.

The third type of data being reported is response time - how long does a caller spend waiting for a request sent over a connection to be processed and responded to by the remote endpoint. The response time reported is an estimation of the true response time of the underlying application protocol. It is computed using heuristics based on the observation of the flow of data between the source and destination end of a physical network connection. Conceptually, it is the difference between the time the last byte of a request leaves the sender, and the time when the last byte of the response arrives back to it. These two timestamps are used to delineate request and response events on a given physical connection. The difference between them represents the response time of a single request.

In this first release of this feature, our algorithm is an approximation that may work with varying degree of success depending on the actual application protocol used for a given network connection. For example, the current approach works well for request-response based protocols such as HTTP(S), but does not work with one-way or message queue-based protocols.

Here are some important points to consider:

1. If a process accepts connections on the same IP address but over multiple network interfaces, a separate record for each interface will be reported.
2. Records with wildcard IP will contain no activity. They are included to represent the fact that a port on the machine is open to inbound traffic.
3. To reduce verbosity and data volume, records with wildcard IP will be omitted when there is a matching record (for the same process, port, and protocol) with a specific IP address. When a wildcard IP record is omitted, the IsWildcardBind record property with the specific IP address, will be set to "True" to indicate that the port is exposed over every interface of the reporting machine.
4. Ports that are bound only on a specific interface have IsWildcardBind set to *False*.

#### Naming and Classification

For convenience, the IP address of the remote end of a connection is included in the Remotelp property. For inbound connections, Remotelp is the same as Sourcelp, while for outbound connections, it is the same as Destinationlp. The RemoteDnsCanonicalNames property represents the DNS canonical names reported by the machine for Remotelp. The RemoteDnsQuestions property represents the DNS questions reported by the machine for Remotelp. The RemoveClassification property is reserved for future use.

#### Geolocation

*VMConnection* also includes geolocation information for the remote end of each connection record in the following properties of the record:

PROPERTY	DESCRIPTION
RemoteCountry	The name of the country/region hosting Remotelp. For example, <i>United States</i>
RemoteLatitude	The geolocation latitude. For example, <i>47.68</i>
RemoteLongitude	The geolocation longitude. For example, <i>-122.12</i>

#### Malicious IP

Every Remotelp property in *VMConnection* table is checked against a set of IPs with known malicious activity. If the Remotelp is identified as malicious the following properties will be populated (they are empty, when the IP is not considered malicious) in the following properties of the record:

PROPERTY	DESCRIPTION
MaliciousIp	The Remotelp address
IndicatorThreadType	Threat indicator detected is one of the following values, <i>Botnet, C2, CryptoMining, Darknet, DDos, MaliciousUrl, Malware, Phishing, Proxy, PUA, Watchlist</i> .
Description	Description of the observed threat.
TLPLevel	Traffic Light Protocol (TLP) Level is one of the defined values, <i>White, Green, Amber, Red</i> .
Confidence	Values are <i>0 – 100</i> .
Severity	Values are <i>0 – 5</i> , where <i>5</i> is the most severe and <i>0</i> is not severe at all. Default value is <i>3</i> .
FirstReportedDateTime	The first time the provider reported the indicator.
LastReportedDateTime	The last time the indicator was seen by Interflow.
IsActive	Indicates indicators are deactivated with <i>True</i> or <i>False</i> value.
ReportReferenceLink	Links to reports related to a given observable.
AdditionalInformation	Provides additional information, if applicable, about the observed threat.

#### Ports

Ports on a machine that actively accept incoming traffic or could potentially accept traffic, but are idle during the reporting time window, are written to the *VMBoundPort* table.

Every record in *VMBoundPort* is identified by the following fields:

PROPERTY	DESCRIPTION
Process	Identity of process (or groups of processes) with which the port is associated with.

PROPERTY	DESCRIPTION
Ip	Port IP address (can be wildcard IP, <i>0.0.0.0</i> )
Port	The Port number
Protocol	The protocol. Example, <i>tcp</i> or <i>udp</i> (only <i>tcp</i> is currently supported).

The identity a port is derived from the above five fields and is stored in the *PortId* property. This property can be used to quickly find records for a specific port across time.

#### Metrics

Port records include metrics representing the connections associated with them. Currently, the following metrics are reported (the details for each metric are described in the previous section):

- BytesSent and BytesReceived
- LinksEstablished, LinksTerminated, LinksLive
- ResponseTime, ResponseTimeMin, ResponseTimeMax, ResponseTimeSum

Here are some important points to consider:

- If a process accepts connections on the same IP address but over multiple network interfaces, a separate record for each interface will be reported.
- Records with wildcard IP will contain no activity. They are included to represent the fact that a port on the machine is open to inbound traffic.
- To reduce verbosity and data volume, records with wildcard IP will be omitted when there is a matching record (for the same process, port, and protocol) with a specific IP address. When a wildcard IP record is omitted, the *IsWildcardBind* property for the record with the specific IP address, will be set to *True*. This indicates the port is exposed over every interface of the reporting machine.
- Ports that are bound only on a specific interface have *IsWildcardBind* set to *False*.

#### VMComputer records

Records with a type of *VMComputer* have inventory data for servers with the Dependency agent. These records have the properties in the following table:

PROPERTY	DESCRIPTION
TenantId	The unique identifier for the workspace
SourceSystem	<i>Insights</i>
TimeGenerated	Timestamp of the record (UTC)
Computer	The computer FQDN
AgentId	The unique ID of the Log Analytics agent
Machine	Name of the Azure Resource Manager resource for the machine exposed by ServiceMap. It is of the form <i>m-{GUID}</i> , where <i>GUID</i> is the same GUID as AgentId.
DisplayName	Display name

PROPERTY	DESCRIPTION
FullDisplayName	Full display name
HostName	The name of machine without domain name
BootTime	The machine boot time (UTC)
TimeZone	The normalized time zone
VirtualizationState	<i>virtual, hypervisor, physical</i>
Ipv4Addresses	Array of IPv4 addresses
Ipv4SubnetMasks	Array of IPv4 subnet masks (in the same order as Ipv4Addresses).
Ipv4DefaultGateways	Array of IPv4 gateways
Ipv6Addresses	Array of IPv6 addresses
MacAddresses	Array of MAC addresses
DnsNames	Array of DNS names associated with the machine.
DependencyAgentVersion	The version of the Dependency agent running on the machine.
OperatingSystemFamily	<i>Linux, Windows</i>
OperatingSystemFullName	The full name of the operating system
PhysicalMemoryMB	The physical memory in megabytes
Cpus	The number of processors
CpuSpeed	The CPU speed in MHz
VirtualMachineType	<i>hyperv, vmware, xen</i>
VirtualMachineNativeId	The VM ID as assigned by its hypervisor
VirtualMachineNativeName	The name of the VM
VirtualMachineHypervisorId	The unique identifier of the hypervisor hosting the VM
HypervisorType	<i>hyperv</i>
HypervisorId	The unique ID of the hypervisor
HostingProvider	<i>azure</i>

PROPERTY	DESCRIPTION
_ResourceId	The unique identifier for an Azure resource
AzureSubscriptionId	A globally unique identifier that identifies your subscription
AzureResourceGroup	The name of the Azure resource group the machine is a member of.
AzureResourceName	The name of the Azure resource
AzureLocation	The location of the Azure resource
AzureUpdateDomain	The name of the Azure update domain
AzureFaultDomain	The name of the Azure fault domain
AzureVmId	The unique identifier of the Azure virtual machine
AzureSize	The size of the Azure VM
AzureImagePublisher	The name of the Azure VM publisher
AzureImageOffering	The name of the Azure VM offer type
AzureImageSku	The SKU of the Azure VM image
AzureImageVersion	The version of the Azure VM image
AzureCloudServiceName	The name of the Azure cloud service
AzureCloudServiceDeployment	Deployment ID for the Cloud Service
AzureCloudServiceRoleName	Cloud Service role name
AzureCloudServiceRoleType	Cloud Service role type: <i>worker</i> or <i>web</i>
AzureCloudServiceInstanceId	Cloud Service role instance ID
AzureVmScaleSetName	The name of the virtual machine scale set
AzureVmScaleSetDeployment	Virtual machine scale set deployment ID
AzureVmScaleSetResourceId	The unique identifier of the virtual machine scale set resource.
AzureVmScaleSetInstanceId	The unique identifier of the virtual machine scale set
AzureServiceFabricClusterId	The unique identifier of the Azure Service Fabric cluster
AzureServiceFabricClusterName	The name of the Azure Service Fabric cluster

## VM Process records

Records with a type of *VMProcess* have inventory data for TCP-connected processes on servers with the Dependency agent. These records have the properties in the following table:

PROPERTY	DESCRIPTION
TenantId	The unique identifier for the workspace
SourceSystem	<i>Insights</i>
TimeGenerated	Timestamp of the record (UTC)
Computer	The computer FQDN
AgentId	The unique ID of the Log Analytics agent
Machine	Name of the Azure Resource Manager resource for the machine exposed by ServiceMap. It is of the form <i>m-{GUID}</i> , where <i>GUID</i> is the same GUID as AgentId.
Process	The unique identifier of the Service Map process. It is in the form of <i>p-{GUID}</i> .
ExecutableName	The name of the process executable
DisplayName	Process display name
Role	Process role: <i>webserver, appServer, databaseServer, ldapServer, smbServer</i>
Group	Process group name. Processes in the same group are logically related, e.g., part of the same product or system component.
StartTime	The process pool start time
FirstPid	The first PID in the process pool
Description	The process description
CompanyName	The name of the company
InternalName	The internal name
ProductName	The name of the product
ProductVersion	The version of the product
FileVersion	The version of the file
ExecutablePath	The path of the executable
CommandLine	The command line
WorkingDirectory	The working directory

PROPERTY	DESCRIPTION
Services	An array of services under which the process is executing
UserName	The account under which the process is executing
UserDomain	The domain under which the process is executing
_ResourceId	The unique identifier for a process within the workspace

## Sample map queries

### List all known machines

```
VMComputer | summarize arg_max(TimeGenerated, *) by _ResourceId
```

### When was the VM last rebooted

```
let Today = now(); VMComputer | extend DaysSinceBoot = Today - BootTime | summarize by Computer, DaysSinceBoot, BootTime | sort by BootTime asc
```

### Summary of Azure VMs by image, location, and SKU

```
VMComputer | where AzureLocation != "" | summarize by Computer, AzureImageOffering, AzureLocation, AzureImageSku
```

### List the physical memory capacity of all managed computers

```
VMComputer | summarize arg_max(TimeGenerated, *) by _ResourceId | project PhysicalMemoryMB, Computer
```

### List computer name, DNS, IP, and OS

```
VMComputer | summarize arg_max(TimeGenerated, *) by _ResourceId | project Computer, OperatingSystemFullName, DnsNames, Ipv4Addresses
```

### Find all processes with "sql" in the command line

```
VMProcess | where CommandLine contains_cs "sql" | summarize arg_max(TimeGenerated, *) by _ResourceId
```

### Find a machine (most recent record) by resource name

```
search in (VMComputer) "m-4b9c93f9-bc37-46df-b43c-899ba829e07b" | summarize arg_max(TimeGenerated, *) by _ResourceId
```

### Find a machine (most recent record) by IP address

```
search in (VMComputer) "10.229.243.232" | summarize arg_max(TimeGenerated, *) by _ResourceId
```

### List all known processes on a specified machine

```
VMProcess | where Machine == "m-559dbcd8-3130-454d-8d1d-f624e57961bc" | summarize arg_max(TimeGenerated, *) by _ResourceId
```

## List all computers running SQL Server

```
VMComputer | where AzureResourceName in ((search in (VMProcess) "*sql*" | distinct Machine)) | distinct Computer
```

## List all unique product versions of curl in my datacenter

```
VMProcess | where ExecutableName == "curl" | distinct ProductVersion
```

## Create a computer group of all computers running CentOS

```
VMComputer | where OperatingSystemFullName contains_cs "CentOS" | distinct Computer
```

## Bytes sent and received trends

```
VMConnection | summarize sum(BytesSent), sum(BytesReceived) by bin(TimeGenerated,1hr), Computer | order by Computer desc | render timechart
```

## Which Azure VMs are transmitting the most bytes

```
VMConnection | join kind=fullouter(VMComputer) on $left.Computer == $right.Computer | summarize count(BytesSent) by Computer, AzureVMSize | sort by count(BytesSent) desc
```

## Link status trends

```
VMConnection | where TimeGenerated >= ago(24hr) | where Computer == "acme-demo" | summarize dcount(LinksEstablished), dcount(LinksLive), dcount(LinksFailed), dcount(LinksTerminated) by bin(TimeGenerated, 1h) | render timechart
```

## Connection failures trend

```
VMConnection | where Computer == "acme-demo" | extend bythehour = datetime_part("hour", TimeGenerated) | project bythehour, LinksFailed | summarize failCount = count() by bythehour | sort by bythehour asc | render timechart
```

## Bound Ports

```
VMBoundPort  
| where TimeGenerated >= ago(24hr)  
| where Computer == 'admdemo-appsvr'  
| distinct Port, ProcessName
```

## Number of open ports across machines

```
VMBoundPort  
| where Ip != "127.0.0.1"  
| summarize by Computer, Machine, Port, Protocol  
| summarize OpenPorts=count() by Computer, Machine  
| order by OpenPorts desc
```

## Score processes in your workspace by the number of ports they have open

```
VMBoundPort
| where Ip != "127.0.0.1"
| summarize by ProcessName, Port, Protocol
| summarize OpenPorts=count() by ProcessName
| order by OpenPorts desc
```

## Aggregate behavior for each port

This query can then be used to score ports by activity, e.g., ports with most inbound/outbound traffic, ports with most connections

```
//
VMBoundPort
| where Ip != "127.0.0.1"
| summarize BytesSent=sum(BytesSent), BytesReceived=sum(BytesReceived),
LinksEstablished=sum(LinksEstablished), LinksTerminated=sum(LinksTerminated), arg_max(TimeGenerated,
LinksLive) by Machine, Computer, ProcessName, Ip, Port, IsWildcardBind
| project-away TimeGenerated
| order by Machine, Computer, Port, Ip, ProcessName
```

## Summarize the outbound connections from a group of machines

```

// the machines of interest
let machines = datatable(m: string) ["m-82412a7a-6a32-45a9-a8d6-538354224a25"];
// map of ip to monitored machine in the environment
let ips=materialize(VMComputer
| summarize ips=makeset(todynamic(Ipv4Addresses)) by MonitoredMachine=AzureResourceName
| mvexpand ips to typeof(string));
// all connections to/from the machines of interest
let out=materialize(VMConnection
| where Machine in (machines)
| summarize arg_max(TimeGenerated, *) by ConnectionId;
// connections to localhost augmented with RemoteMachine
let local=out
| where RemoteIp startswith "127."
| project ConnectionId, Direction, Machine, Process, ProcessName, SourceIp, DestinationIp, DestinationPort,
Protocol, RemoteIp, RemoteMachine=Machine;
// connections not to localhost augmented with RemoteMachine
let remote=materialize(out
| where RemoteIp !startswith "127."
| join kind=leftouter (ips) on $left.RemoteIp == $right.ips
| summarize by ConnectionId, Direction, Machine, Process, ProcessName, SourceIp, DestinationIp,
DestinationPort, Protocol, RemoteIp, RemoteMachine=MonitoredMachine);
// the remote machines to/from which we have connections
let remoteMachines = remote | summarize by RemoteMachine;
// all augmented connections
(local)
| union (remote)
//Take all outbound records but only inbound records that come from either //unmonitored machines or
monitored machines not in the set for which we are computing dependencies.
| where Direction == 'outbound' or (Direction == 'inbound' and RemoteMachine !in (machines))
| summarize by ConnectionId, Direction, Machine, Process, ProcessName, SourceIp, DestinationIp,
DestinationPort, Protocol, RemoteIp, RemoteMachine
// identify the remote port
| extend RemotePort=iff(Direction == 'outbound', DestinationPort, 0)
// construct the join key we'll use to find a matching port
| extend JoinKey=strcat_delim(':', RemoteMachine, RemoteIp, RemotePort, Protocol)
// find a matching port
| join kind=leftouter (VMBoundPort
| where Machine in (remoteMachines)
| summarize arg_max(TimeGenerated, *) by PortId
| extend JoinKey=strcat_delim(':', Machine, Ip, Port, Protocol)) on JoinKey
// aggregate the remote information
| summarize Remote=makeset(ifempty(RemoteMachine), todynamic('{}'), pack('Machine', RemoteMachine,
'Process', Process1, 'ProcessName', ProcessName1))) by ConnectionId, Direction, Machine, Process,
ProcessName, SourceIp, DestinationIp, DestinationPort, Protocol

```

## Performance records

Records with a type of *InsightsMetrics* have performance data from the guest operating system of the virtual machine. These records have the properties in the following table:

PROPERTY	DESCRIPTION
TenantId	Unique identifier for the workspace
SourceSystem	<i>Insights</i>
TimeGenerated	Time the value was collected (UTC)
Computer	The computer FQDN
Origin	<i>vm.azm.ms</i>

PROPERTY	DESCRIPTION
Namespace	Category of the performance counter
Name	Name of the performance counter
Val	Collected value
Tags	Related details about the record. See the table below for tags used with different record types.
AgentId	Unique identifier for each computer's agent
Type	<i>InsightsMetrics</i>
ResourceId	Resource ID of the virtual machine

The performance counters currently collected into the *InsightsMetrics* table are listed in the following table:

NAMESPACE	NAME	DESCRIPTION	UNIT	TAGS
Computer	Heartbeat	Computer Heartbeat		
Memory	AvailableMB	Memory Available Bytes	Megabytes	memorySizeMB - Total memory size
Network	WriteBytesPerSecond	Network Write Bytes Per Second	BytesPerSecond	NetworkDeviceId - Id of the device bytes - Total sent bytes
Network	ReadBytesPerSecond	Network Read Bytes Per Second	BytesPerSecond	networkDeviceId - Id of the device bytes - Total received bytes
Processor	UtilizationPercentage	Processor Utilization Percentage	Percent	totalCpus - Total CPUs
LogicalDisk	WritesPerSecond	Logical Disk Writes Per Second	CountPerSecond	mountId - Mount ID of the device
LogicalDisk	WriteLatencyMs	Logical Disk Write Latency Millisecond	Milliseconds	mountId - Mount ID of the device
LogicalDisk	WriteBytesPerSecond	Logical Disk Write Bytes Per Second	BytesPerSecond	mountId - Mount ID of the device
LogicalDisk	TransfersPerSecond	Logical Disk Transfers Per Second	CountPerSecond	mountId - Mount ID of the device
LogicalDisk	TransferLatencyMs	Logical Disk Transfer Latency Millisecond	Milliseconds	mountId - Mount ID of the device

NAMESPACE	NAME	DESCRIPTION	UNIT	TAGS
LogicalDisk	ReadsPerSecond	Logical Disk Reads Per Second	CountPerSecond	mountId - Mount ID of the device
LogicalDisk	ReadLatencyMs	Logical Disk Read Latency Millisecond	MilliSeconds	mountId - Mount ID of the device
LogicalDisk	ReadBytesPerSecond	Logical Disk Read Bytes Per Second	BytesPerSecond	mountId - Mount ID of the device
LogicalDisk	FreeSpacePercentage	Logical Disk Free Space Percentage	Percent	mountId - Mount ID of the device
LogicalDisk	FreeSpaceMB	Logical Disk Free Space Bytes	Megabytes	mountId - Mount ID of the device diskSizeMB - Total disk size
LogicalDisk	BytesPerSecond	Logical Disk Bytes Per Second	BytesPerSecond	mountId - Mount ID of the device

## Next steps

- If you are new to writing log queries in Azure Monitor, review [how to use Log Analytics](#) in the Azure portal to write log queries.
- Learn about [writing search queries](#).

# Create interactive reports VM insights with workbooks

9/21/2022 • 12 minutes to read • [Edit Online](#)

Workbooks combine text, log queries, metrics, and parameters into rich interactive reports. Workbooks are editable by any other team members who have access to the same Azure resources.

Workbooks are helpful for scenarios such as:

- Exploring the usage of your virtual machine when you don't know the metrics of interest in advance: CPU utilization, disk space, memory, network dependencies, etc. Unlike other usage analytics tools, workbooks let you combine multiple kinds of visualizations and analyses, making them great for this kind of free-form exploration.
- Explaining to your team how a recently provisioned VM is performing, by showing metrics for key counters and other log events.
- Sharing the results of a resizing experiment of your VM with other members of your team. You can explain the goals for the experiment with text, then show each usage metric and analytics queries used to evaluate the experiment, along with clear call-outs for whether each metric was above- or below-target.
- Reporting the impact of an outage on the usage of your VM, combining data, text explanation, and a discussion of next steps to prevent outages in the future.

## VM insights workbooks

VM insights includes the following workbooks. You can use these workbooks or use them as a start to create custom workbooks to address your particular requirements.

### Single virtual machine

WORKBOOK	DESCRIPTION
Performance	Provides a customizable version of the Performance view that leverages all of the Log Analytics performance counters that you have enabled.
Connections	Connections provides an in-depth view of the inbound and outbound connections from your VM.

### Multiple virtual machines

WORKBOOK	DESCRIPTION
Performance	Provides a customizable version of the Top N List and Charts view in a single workbook that leverages all of the Log Analytics performance counters that you have enabled.
Performance counters	A Top N chart view across a wide set of performance counters.
Connections	Connections provides an in-depth view of the inbound and outbound connections from your monitored VMs.

WORKBOOK	DESCRIPTION
Active Ports	Provides a list of the processes that have bound to the ports on the monitored VMs and their activity in the chosen timeframe.
Open Ports	Provides the number of ports open on your monitored VMs and the details on those open ports.
Failed Connections	Display the count of failed connections on your monitored VMs, the failure trend, and if the percentage of failures is increasing over time.
Security and Audit	An analysis of your TCP/IP traffic that reports on overall connections, malicious connections, where the IP endpoints reside globally. To enable all features, you will need to enable Security Detection.
TCP Traffic	A ranked report for your monitored VMs and their sent, received, and total network traffic in a grid and displayed as a trend line.
Traffic Comparison	This workbook lets you compare network traffic trends for a single machine or a group of machines.

## Creating a new workbook

A workbook is made up of sections consisting of independently editable charts, tables, text, and input controls. To better understand workbooks, let's start by opening a template and walk through creating a custom workbook.

1. Go to the **Monitor** menu in the Azure portal.
2. Select a virtual machine.
3. On the VM insights page, select **Performance** or **Maps** tab and then select **View Workbooks** from the link on the page. From the drop-down list, select **Go to Gallery**.

The screenshot shows the Azure VM Insights interface for a virtual machine named 'ContosoWeb1'. The 'Performance' tab is selected. In the center, there are three charts: 'Logical Disk Performance' (showing disk usage for C:, D:, E:, and Total), 'CPU Utilization %' (a line chart showing CPU usage over time), and 'Available Memory' (a line chart showing memory usage over time). On the right side, there is a sidebar with various monitoring links like 'Resource Group Monitoring', 'Azure Monitor', 'Run Diagnostics', and 'Properties'. The 'Workbooks' section is expanded, showing 'View Workbooks' (dropdown), 'Performance Analysis', 'Network Dependencies', and 'Workbooks Gallery'. A red box highlights the 'Workbooks Gallery' link. At the bottom of the sidebar, there are links for 'Log Events' and 'Alerts'.

This launches the workbook gallery with a number of prebuilt workbooks to help you get started.

#### 4. Create a new workbook by selecting New.

The screenshot shows the Azure Monitor Gallery interface. At the top, there are navigation links for Home, Monitor, and a search bar. Below that, a header bar includes 'New', 'Refresh', 'Feedback', 'Help', 'Community Git repo', 'Browse across galleries', and filter options for 'Subscription: AzureMonitor\_Docs' and 'Resource Group: All'. A 'Reset filters' button is also present. The main content area is organized into sections: 'Quick start' (Empty), 'Recently modified workbooks (3)' (VM Insights Report 3, VM Insights Report 2, VM Insights Report 1), 'Network Dependencies (7)' (Connections Overview, Open Ports, Active Ports, Failed Connections, Security and Audit, Traffic Comparison, TCP Traffic), and 'Performance Analysis (2)' (Performance, Perf Counters). Each item has a thumbnail icon and a brief description.

## Editing workbook sections

Workbooks have two modes: **editing mode**, and **reading mode**. When a new workbook is first launched, it opens in **editing mode**. It shows all the content of the workbook, including any steps and parameters that are otherwise hidden. **Reading mode** presents a simplified report style view. Reading mode allows you to abstract away the complexity that went into creating a report while still having the underlying mechanics only a few clicks away when needed for modification.

The screenshot shows a 'New workbook' page in the Azure Monitor. The top navigation bar includes 'Home > Monitor - Virtual Machines > Workbook 1', 'Workbook 1', 'InfraLabDefaultWorkspace', and standard browser controls. Below the toolbar, a section titled 'New workbook' contains a welcome message and a basic analytics query: `union withsource=TableName * | summarize Count=count() by TableName | render barchart`. The 'Edit' button is visible in the top right of this section. At the bottom of the section, there is a note about a failed operation: 'Can not perform requested operation on nested resource. Parent resource 'InfraLabDefaultWorkspace' not found. Click to Retry.' Below this note are four numbered buttons (1, 2, 3, 4) and a row of action buttons: 'Done Editing', 'Add text', 'Add query', 'Add metric', 'Add parameters', and icons for copy/paste and refresh. A footer at the bottom of the page provides links to 'Add text', 'Add query', 'Add metric', and 'Add parameters'.

1. When you're done editing a section, click **Done Editing** in the bottom-left corner of the section.
2. To create a duplicate of a section, click the **Clone this section** icon. Creating duplicate sections is a great way to iterate on a query without losing previous iterations.
3. To move up a section in a workbook, click the **Move up** or **Move down** icon.
4. To remove a section permanently, click the **Remove** icon.

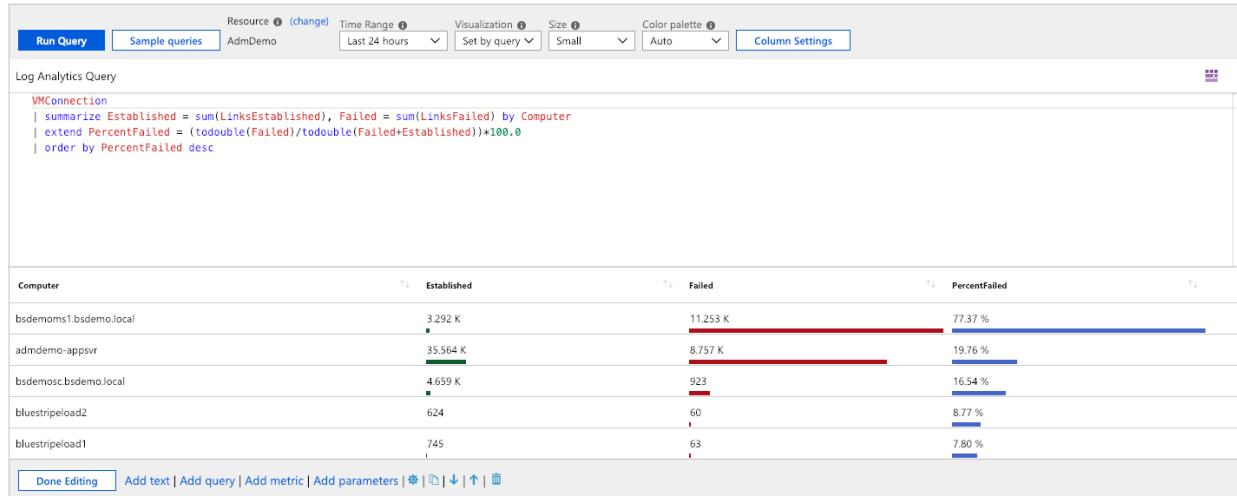
## Adding text and Markdown sections

Adding headings, explanations, and commentary to your workbooks helps turn a set of tables and charts into a

narrative. Text sections in workbooks support the [Markdown syntax](#) for text formatting, like headings, bold, italics, and bulleted lists.

To add a text section to your workbook, use the **Add text** button at the bottom of the workbook, or at the bottom of any section.

## Adding query sections



To add query section to your workbook, use the **Add query** button at the bottom of the workbook, or at the bottom of any section.

Query sections are highly flexible and can be used to answer questions like:

- How was my CPU utilization during the same time period as an increase in network traffic?
- What was the trend in available disk space over the last month?
- How many network connection failures did my VM experience over the last two weeks?

You also aren't only limited to querying from the context of the virtual machine you launched the workbook from. You can query across multiple virtual machines, as well as Log Analytics workspaces, as long as you have access permission to those resources.

To include data from other Log Analytics workspaces or from a specific Application Insights app using the **workspace** identifier. To learn more about cross-resource queries, refer to the [official guidance](#).

### Advanced analytic query settings

Each section has its own advanced settings, which are accessible via the settings icon located to the right of the **Add parameters** button.

The screenshot shows the 'Advanced Settings' dialog for a query section. It contains the following options:

- Make this item a custom width [i](#)
- Make this item conditionally visible [i](#)
- When an item is selected, export a parameter [i](#)
- Show query when not editing
- Show open in analytics button when not editing

Below these is a 'Chart title' field with the placeholder 'Enter a title, or leave blank for no title'.

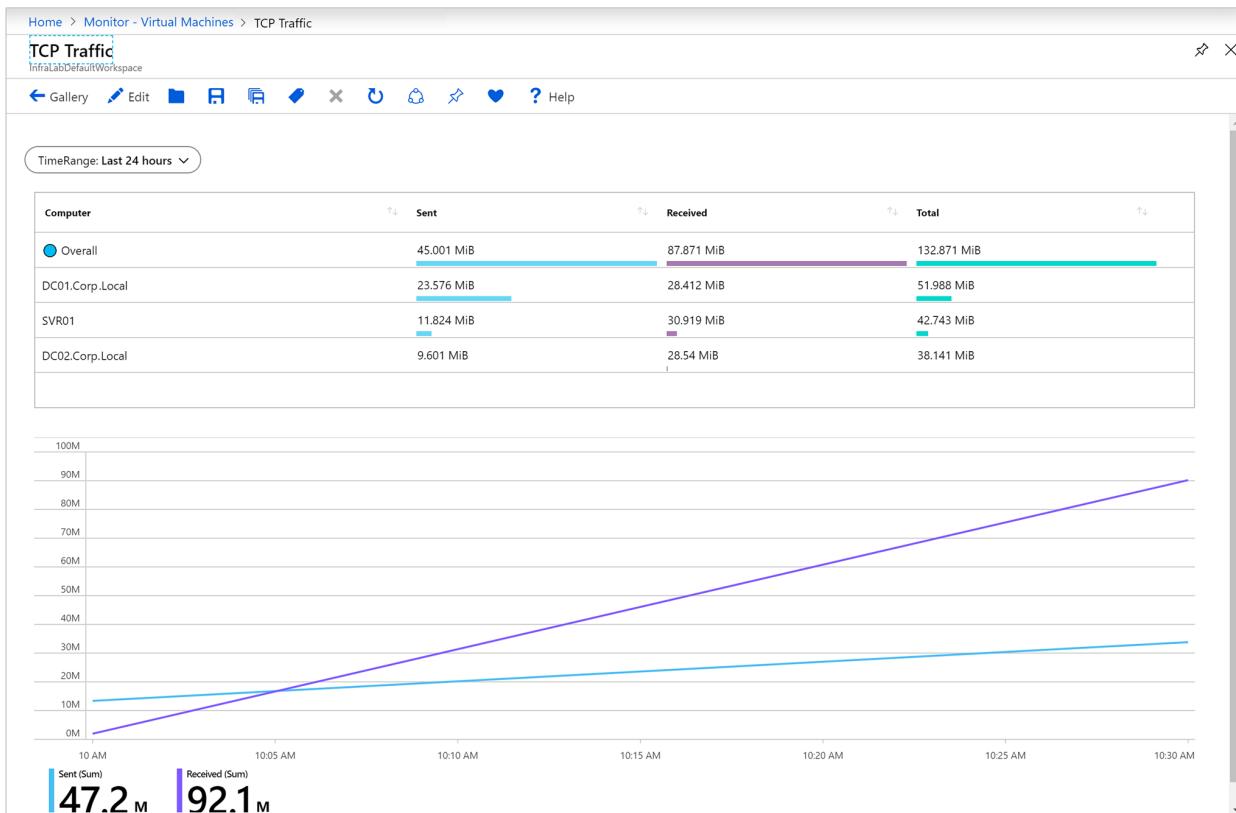
At the bottom are buttons for 'Done Editing', 'Add text', 'Add query', 'Add metric', 'Add parameters', and icons for copy, paste, up, down, and delete.

SETTING	DESCRIPTION
Custom width	Makes an item an arbitrary size, so you can fit many items on a single line allowing you to better organize your charts and tables into rich interactive reports.
Conditionally visible	Specify to hide steps based on a parameter when in reading mode.
Export a parameter	Allow a selected row in the grid or chart to cause later steps to change values or become visible.
Show query when not editing	Displays the query above the chart or table even when in reading mode.
Show open in analytics button when not editing	Adds the blue Analytics icon to the right-hand corner of the chart to allow one-click access.

Most of these settings are fairly intuitive, but to understand **Export a parameter** it is better to examine a workbook that makes use of this functionality.

One of the prebuilt workbooks - **TCP Traffic**, provides information on connection metrics from a VM.

The first section of the workbook is based on log query data. The second section is also based on log query data, but selecting a row in the first table will interactively update the contents of the charts:



The behavior is possible through use of the **When an item is selected, export a parameter** advanced settings, which are enabled in the table's log query.

## Advanced Settings

- Make this item a custom width [i](#)
- Make this item conditionally visible [i](#)
- When an item is selected, export a parameter [i](#)

Field to export [i](#)

Values

Parameter name

Filter

The second log query then utilizes the exported values when a row is selected to create a set of values that are then used by the section heading and charts. If no row is selected, it hides the section heading and charts.

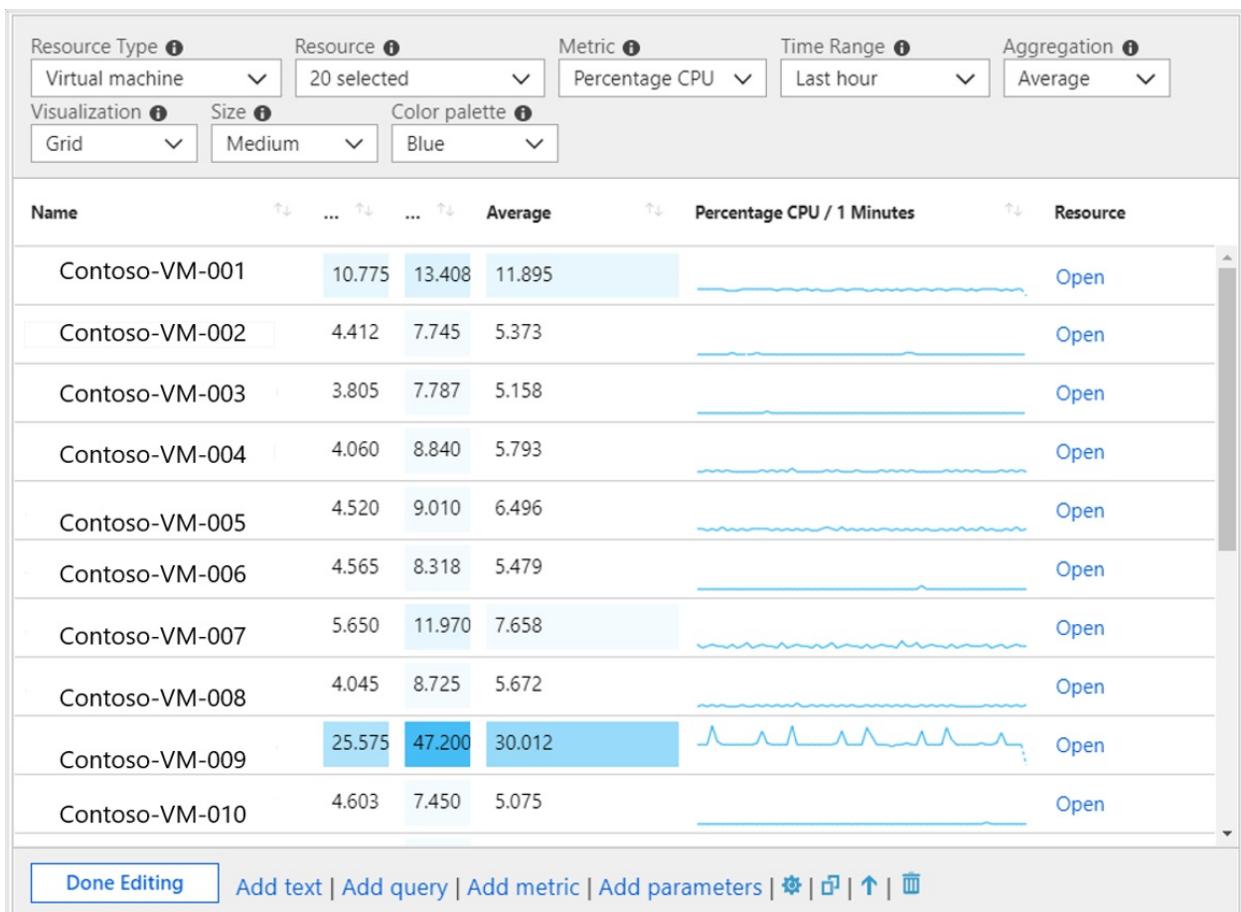
For example, the hidden parameter in the second section uses the following reference from the row selected in the grid:

```
VMConnection
| where TimeGenerated {TimeRange}
| where Computer in ("{ComputerName}") or '*' in ("{ComputerName}")
| summarize Sent = sum(BytesSent), Received = sum(BytesReceived) by bin(TimeGenerated, {TimeRange:grain})
```

## Adding metrics sections

Metrics sections give you full access to incorporate Azure Monitor metrics data into your interactive reports. In VM insights, the prebuilt workbooks will typically contain analytic query data rather than metric data. You may choose to create workbooks with metric data, allowing you to take full advantage of the best of both features all in one place. You also have the ability to pull in metric data from resources in any of the subscriptions you have access to.

Here is an example of virtual machine data being pulled into a workbook to provide a grid visualization of CPU performance:



## Adding parameter sections

Workbook parameters allow you to change values in the workbook without having to manually edit the query or text sections. This removes the requirement of needing to understand the underlying analytics query language and greatly expands the potential audience of workbook-based reporting.

The values of parameters are replaced in query, text or other parameter sections by putting the name of the parameter in braces, like `{parameterName}`. Parameter names are limited to similar rules as JavaScript identifiers, alphabetic characters or underscores, followed by alphanumeric characters or underscores. For example, `a1` is allowed, but `1a` is not allowed.

Parameters are linear, starting from the top of a workbook and flowing down to later steps. Parameters declared later in a workbook can override parameters that were declared earlier. This also lets parameters that use queries to access the values from parameters defined earlier. Within a parameter's step itself, parameters are also linear, left to right, where parameters to the right can depend on a parameter declared earlier in that same step.

There are four different types of parameters, which are currently supported:

PARAMETER	DESCRIPTION
Text	Allows the user to edit a text box, and you can optionally supply a query to fill in the default value.
Drop down	Allows the user to choose from a set of values.
Time range picker	Allows the user to choose from a predefined set of time range values, or pick from a custom time range.

PARAMETER	DESCRIPTION
Resource picker	Allows the user to choose from the resources selected for the workbook.

## Using a text parameter

The value a user types in the text box is replaced directly in the query, with no escaping or quoting. If the value you need is a string, the query should have quotes around the parameter (like '{parameter}').

The text parameter allows the value in a text box to be used anywhere. It can be a table name, column name, function name, operator, etc. The text parameter type has a setting **Get default value from analytics query**, which allows the workbook author to use a query to populate the default value for that text box.

When using the default value from a log query, only the first value of the first row (row 0, column 0) is used as the default value. Therefore it is recommended to limit your query to return just one row and one column. Any other data returned by the query is ignored.

Whatever value the query returns will be replaced directly with no escaping or quoting. If the query returns no rows, the result of the parameter is either an empty string (if the parameter is not required) or undefined (if the parameter is required).

## Using a drop-down

The dropdown parameter type lets you create a drop-down control, allowing the selection of one or many values.

The dropdown is populated by a log query or JSON. If the query returns one column, the values in that column are both the value and the label in the drop-down control. If the query returns two columns, the first column is the value, and the second column is the label shown in the drop-down. If the query returns three columns, the third column is used to indicate the default selection in that drop-down. This column can be any type, but the simplest is to use bool or numeric types, where 0 is false, and 1 is true.

If the column is a string type, null/empty string is considered false, and any other value is considered true. For single selection drop-downs, the first value with a true value is used as the default selection. For multiple selection drop-downs, all values with a true value are used as the default selected set. The items in the drop-down are shown in whatever order the query returned rows.

Let's look at the parameters present in the Connections Overview report. Click the edit symbol next to **Direction**.



This will launch the **Edit Parameter** menu item.

**Edit Parameter**

AdmDemo

Save Revert changes Cancel Help

* Parameter name ⓘ	Direction
Parameter type ⓘ	Drop down
Required? ⓘ	<input checked="" type="checkbox"/>
Allow multiple selections ⓘ	<input type="checkbox"/>
Limit multiple selections ⓘ	<input type="checkbox"/>
Delimiter ⓘ	,
Quote with ⓘ	'
Explanation ⓘ	Direction of the network connection from the VMs
Hide parameter in reading mode ⓘ	<input type="checkbox"/>
Get data from ⓘ	<a href="#">Query</a> <a href="#">JSON</a>

JSON Input ⓘ

Update

```
[{"value": "inbound", "label": "Inbound"}, {"value": "outbound", "label": "Outbound"}]
```

The JSON lets you generate an arbitrary table populated with content. For example, the following JSON generates two values in the drop-down:

```
[{"value": "inbound", "label": "Inbound"}, {"value": "outbound", "label": "Outbound"}]
```

A more applicable example is using a drop-down to pick from a set of performance counters by name:

```
Perf
| summarize by CounterName, ObjectName
| order by ObjectName asc, CounterName asc
| project Counter = pack('counter', CounterName, 'object', ObjectName), CounterName, group = ObjectName
```

The query will display results as follows:

## Edit Parameter

AdmDemo

Save Revert changes Cancel Help

* Parameter name ⓘ	Counter
Parameter type ⓘ	Drop down
Required? ⓘ	<input checked="" type="checkbox"/>
Allow multiple selections ⓘ	<input type="checkbox"/>
Limit multiple selections ⓘ	<input type="checkbox"/>
Delimiter ⓘ	,
Quote with ⓘ	'
Explanation ⓘ	Select a VM performance counter for the table below
Hide parameter in reading mode ⓘ	<input type="checkbox"/>
Get data from ⓘ	<b>Query</b> JSON

Log Analytics Query ⓘ

Resource ⓘ (change) Time Range ⓘ

**Run Query** AdmDemo Set in query Samples

```
// {Workspaces:label}
Perf
| where TimeGenerated < TimeRange
| where ObjectName != 'Network' and ObjectName != 'Network Interface'
| summarize by CounterName, ObjectName, CounterText = CounterName
| order by ObjectName asc, CounterText asc
| project Counter = pack('counter', CounterName, 'object', ObjectName)
, CounterText, group = ObjectName
```

Counter	CounterText	group
("counter": "% Used Inodes", "object": "Logical Disk")	% Used Inodes	Log
("counter": "% Used Space", "object": "Logical Disk")	% Used Space	Log
("counter": "Disk Read Bytes/sec", "object": "Logical Disk")	Disk Read Bytes/sec	Log
("counter": "Disk Reads/sec", "object": "Logical Disk")	Disk Reads/sec	Log

Drop-downs are incredibly powerful tools for customizing and creating interactive reports.

### Time range parameters

While you can make your own custom time range parameter via the dropdown parameter type, you can also use the out-of-box time range parameter type if you don't need the same degree of flexibility.

Time range parameter types have 15 default ranges that go from five minutes to the last 90 days. There is also an option to allow custom time range selection, which allows the operator of the report to choose explicit start and stop values for the time range.

### Resource picker

The resource picker parameter type gives you the ability to scope your report to certain types of resources. An example of a prebuilt workbook that leverages the resource picker type is the **Performance** workbook.

## Edit Parameter



AdmDemo

Save

Revert changes

Cancel

Help

\* Parameter name

Workspaces|

Parameter type

Resource picker



Required?



Allow multiple selections



Limit multiple selections



Delimiter



Quote with



Explanation

*What is this parameter used for?*

Hide parameter in reading mode



Get data from

Workbook Resources

Query

JSON

Include in the drop down

Any one

Any three

Any five

Any ten

All

Resource Filtering

(Optional) Include only resource types

Log Analytics workspace



(Optional) Include names containing

Previews

When editing, your parameter will look like this:

Workspaces: 0 selected

When not editing, your parameter will look like this:

## Saving and sharing workbooks with your team

Workbooks are saved within a Log Analytics Workspace or a virtual machine resource, depending on how you access the workbooks gallery. The workbook can be saved to the **My Reports** section that's private to you or in the **Shared Reports** section that's accessible to everyone with access to the resource. To view all the

workbooks in the resource, click the **Open** button in the action bar.

To share a workbook that's currently in **My Reports**:

1. Click **Open** in the action bar
2. Click the "..." button beside the workbook you want to share
3. Click **Move to Shared Reports**.

To share a workbook with a link or via email, click **Share** in the action bar. Keep in mind that recipients of the link need access to this resource in the Azure portal to view the workbook. To make edits, recipients need at least Contributor permissions for the resource.

To pin a link to a workbook to an Azure Dashboard:

1. Click **Open** in the action bar
2. Click the "..." button beside the workbook you want to pin
3. Click **Pin to dashboard**.

## Next steps

- To identify limitations and overall VM performance, see [View Azure VM Performance](#).
- To learn about discovered application dependencies, see [View VM insights Map](#).

# Monitor virtual machines with Azure Monitor: Alerts

9/21/2022 • 10 minutes to read • [Edit Online](#)

This article is part of the scenario [Monitor virtual machines and their workloads in Azure Monitor](#). It provides guidance on creating alert rules for your virtual machines and their guest operating systems. [Alerts in Azure Monitor](#) proactively notify you of interesting data and patterns in your monitoring data. There are no preconfigured alert rules for virtual machines, but you can create your own based on data collected by VM insights.

## NOTE

This scenario describes how to implement complete monitoring of your Azure and hybrid virtual machine environment. To get started monitoring your first Azure virtual machine, see [Monitor Azure virtual machines](#), [Tutorial: Create a metric alert for an Azure resource](#), or [Tutorial: Create alert when Azure virtual machine is unavailable](#).

## IMPORTANT

Most alert rules have a cost that's dependent on the type of rule, how many dimensions it includes, and how frequently it's run. Before you create any alert rules, refer to [Alert rules in Azure Monitor pricing](#).

## Choose the alert type

The most common types of alert rules in Azure Monitor are [metric alerts](#) and [log query alerts](#). The type of alert rule that you create for a particular scenario depends on where the data is located that you're alerting on. You might have cases where data for a particular alerting scenario is available in both Metrics and Logs, and you'll need to determine which rule type to use. You might also have flexibility in how you collect certain data and let your decision of alert rule type drive your decision for data collection method.

Typically, the best strategy is to use metric alerts instead of log alerts when possible because they're more responsive and stateful. To use metric alerts, the data you're alerting on must be available in Metrics. VM insights currently sends all of its data to Logs, so you must install the Azure Monitor agent to use metric alerts with data from the guest operating system. Use Log query alerts with metric data when it's unavailable in Metrics or if you require logic beyond the relatively simple logic for a metric alert rule.

### Metric alerts

[Metric alert rules](#) are useful for alerting when a particular metric exceeds a threshold. An example is when the CPU of a machine is running high. The target of a metric alert rule can be a specific machine, a resource group, or a subscription. In this instance, you can create a single rule that applies to a group of machines.

Metric rules for virtual machines can use the following data:

- Host metrics for Azure virtual machines, which are collected automatically.
- Metrics that are collected by the Azure Monitor agent from the guest operating system.

## NOTE

When VM insights supports the Azure Monitor agent, which is currently in public preview, it sends performance data from the guest operating system to Metrics so that you can use metric alerts.

## Log alerts

Log alerts can measure two different things which can be used to monitor virtual machines in different scenarios:

- **Result count:** Counts the number of rows returned by the query, and can be used to work with events such as Windows event logs, syslog, application exceptions.
- **Calculation of a value:** Makes a calculation based on a numeric column, and can be used to include any number of resources. For example, CPU percentage.

## Targeting resources and dimensions

You can monitor multiple instances' values with one rule using dimensions. You would use dimensions if, for example, you want to monitor CPU usage on multiple instances running your web site or app for CPU usage over 80%.

To create resource-centric alerts at scale for a subscription or resource group, you can **Split by dimensions**. When you want to monitor the same condition on multiple Azure resources, splitting by dimensions splits the alerts into separate alerts by grouping unique combinations using numerical or string columns. Splitting on Azure resource ID column makes the specified resource into the alert target.

You may also decide not to split when you want a condition on multiple resources in the scope, for example, if you want to alert if at least five machines in the resource group scope have CPU usage over 80%.

The screenshot shows the 'Create an alert rule' interface in the Microsoft Azure (Preview) portal. The 'Condition' tab is active. In the 'Split by dimensions' section, a dimension named '\_ResourceId' is selected. The 'Alert logic' section includes an operator 'Less than', a threshold value of '20', and a frequency of evaluation of '10 minutes'. The top right of the interface includes a search bar and a 'Report a bug' button.

You might want to see a list of the alerts with the affected computers. You can use a custom workbook that uses a custom [Resource Graph](#) to provide this view. Use the following query to display alerts, and use the data source

Azure Resource Graph in the workbook.

```
alertsmanagementresources
| extend dimension = properties.context.context.condition.allOf
| mv-expand dimension
| extend dimension = dimension.dimensions
| mv-expand dimension
| extend Computer = dimension.value
| extend AlertStatus = properties.essentials.alertState
| summarize count() by Alert=name, tostring(AlertStatus), tostring(Computer)
| project Alert, AlertStatus, Computer
```

## Common alert rules

The following section lists common alert rules for virtual machines in Azure Monitor. Details for metric alerts and log metric measurement alerts are provided for each. For guidance on which type of alert to use, see [Choose the alert type](#).

If you're unfamiliar with the process for creating alert rules in Azure Monitor, see the [instructions to create a new alert rule](#).

### Machine unavailable

The most basic requirement is to send an alert when a machine is unavailable. It could be stopped, the guest operating system could be unresponsive, or the agent could be unresponsive. There are various ways to configure this alerting, but the most common is to use the heartbeat sent from the Log Analytics agent.

#### Log query alert rules

Log query alerts use the [Heartbeat table](#), which should have a heartbeat record every minute from each machine.

Use a rule with the following query.

```
Heartbeat
| summarize TimeGenerated=max(TimeGenerated) by Computer, _ResourceId
| extend Duration = datetime_diff('minute',now(),TimeGenerated)
| summarize AggregatedValue = min(Duration) by Computer, bin(TimeGenerated,5m), _ResourceId
```

#### Metric alert rules

A metric called *Heartbeat* is included in each Log Analytics workspace. Each virtual machine connected to that workspace sends a heartbeat metric value each minute. Because the computer is a dimension on the metric, you can fire an alert when any computer fails to send a heartbeat. Set the **Aggregation type** to **Count** and the **Threshold** value to match the **Evaluation granularity**.

### CPU alerts

#### Metric alert rules

TARGET	METRIC
Host	Percentage CPU
Windows guest	\Processor Information(_Total)% Processor Time
Linux guest	cpu/usage_active

#### Log alert rules

### CPU utilization

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Processor" and Name == "UtilizationPercentage"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

## Memory alerts

### Metric alert rules

TARGET	METRIC
Windows guest	\Memory% Committed Bytes in Use \Memory\Available Bytes
Linux guest	mem/available mem/available_percent

### Log alert rules

#### Available memory in MB

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Memory" and Name == "AvailableMB"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

#### Available memory in percentage

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Memory" and Name == "AvailableMB"
| extend TotalMemory = toreal(todynamic(Tags)["vm.azm.ms/memorySizeMB"])
| extend AvailableMemoryPercentage = (toreal(Val) / TotalMemory) * 100.0
| summarize AggregatedValue = avg(AvailableMemoryPercentage) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

## Disk alerts

### Metric alert rules

TARGET	METRIC
Windows guest	\Logical Disk(_Total)% Free Space \Logical Disk(_Total)\Free Megabytes
Linux guest	disk/free disk/free_percent

### Log query alert rules

#### Logical disk used - all disks on each computer

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "LogicalDisk" and Name == "FreeSpacePercentage"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

#### Logical disk used - individual disks

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "LogicalDisk" and Name == "FreeSpacePercentage"
| extend Disk=tostring(todynamic(Tags)["vm.azm.ms/mountId"])
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId, Disk

```

## Logical disk IOPS

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "LogicalDisk" and Name == "TransfersPerSecond"
| extend Disk=tostring(todynamic(Tags)["vm.azm.ms/mountId"])
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId, Disk

```

## Logical disk data rate

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "LogicalDisk" and Name == "BytesPerSecond"
| extend Disk=tostring(todynamic(Tags)["vm.azm.ms/mountId"])
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId, Disk

```

# Network alerts

## Metric alert rules

TARGET	METRIC
Windows guest	\Network Interface\Bytes Sent/sec \Logical Disk(_Total)\Free Megabytes
Linux guest	disk/free disk/free_percent

## Log query alert rules

### Network interfaces bytes received - all interfaces

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Network" and Name == "ReadBytesPerSecond"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

### Network interfaces bytes received - individual interfaces

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Network" and Name == "ReadBytesPerSecond"
| extend NetworkInterface=tostring(todynamic(Tags)["vm.azm.ms/networkDeviceId"])
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId, NetworkInterface

```

### Network interfaces bytes sent - all interfaces

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Network" and Name == "WriteBytesPerSecond"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

## Network interfaces bytes sent - individual interfaces

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Network" and Name == "WriteBytesPerSecond"
| extend NetworkInterface=tostring(todynamic(Tags)["vm.azm.ms/networkDeviceId"])
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId, NetworkInterface

```

## Example log query alert

Here's a walk-through of creating a log alert for when the CPU of a virtual machine exceeds 80 percent. The data you need is in the [InsightsMetrics table](#). The following query returns the records that need to be evaluated for the alert. Each type of alert rule uses a variant of this query.

### Create the log alert rule

1. In the portal, select the relevant resource. We recommend scaling resources by using subscriptions or resource groups.
2. In the Resource menu, select **Logs**.
3. Use this query to monitor for virtual machines CPU usage:

```

InsightsMetrics
| where Origin == "vm.azm.ms"
| where Namespace == "Processor" and Name == "UtilizationPercentage"
| summarize AggregatedValue = avg(Val) by bin(TimeGenerated, 15m), Computer, _ResourceId

```

4. Run the query to make sure you get the results you were expecting.
5. From the top command bar, Select **+ New alert rule** to create a rule using the current query.
6. The **Create an alert rule** page opens with your query. We try to detect summarized data from the query results automatically. If detected, the appropriate values are automatically selected.

The screenshot shows the 'Create an alert rule' interface. On the left, there's a sidebar with icons for Scope, Condition, Actions, Details, Tags, and Review + create. The 'Scope' tab is currently selected. The main area has tabs for Scope, Condition, Actions, Details, Tags, and Review + create. Under 'Condition', there's a star icon and a note: 'Configure when the alert rule should trigger by selecting a signal and defining its logic.' Below that is a 'Log query' section with a note: 'Define the logic for triggering an alert. Use the chart to view trends in the data. Learn more'. It shows the query: 'The query to run on this resource's logs. The results returned by this query are used to populate the alert definition below.' A 'Search query' input field contains the InsightsMetrics query. At the bottom of the 'Log query' section is a link: 'View result and edit query in Logs'.

7. In the **Measurement** section, select the values for these fields if they are not already automatically selected.

FIELD	DESCRIPTION	VALUE FOR THIS SCENARIO
-------	-------------	-------------------------

FIELD	DESCRIPTION	VALUE FOR THIS SCENARIO
Measure	The number of table rows or a numeric column to aggregate	AggregatedValue
Aggregation type	The type of aggregation to apply to the data points in aggregation granularity	Average
Aggregation granularity	The interval over which data points are grouped by the aggregation type	15 minutes

Measurement

Select how to summarize the results. We try to detect summarized data from the query results automatically.

Measure: AggregatedValue

Aggregation type: Average

Aggregation granularity: 15 minutes

8. In the **Split by dimensions** section, select the values for these fields if they are not already automatically selected.

FIELD	DESCRIPTION	VALUE FOR THIS SCENARIO
Resource ID column	An Azure Resource ID column that will split the alerts and set the fired alert target scope.	_ResourceId
Dimension name	Dimensions monitor specific time series and provide context to the fired alert. Dimensions can be either number or string columns. If you select more than one dimension value, each time series that results from the combination will trigger its own alert and will be charged separately. The displayed dimension values are based on data from the last 48 hours. Custom dimension values can be added by clicking 'Add custom value'.	Computer
Operator	The operator to compare the dimension value	=
Dimension value	The list of dimension column values	All current and future values

Split by dimensions

Resource ID column: \_ResourceId

Dimension name	Operator	Dimension values
Computer	=	All current and future values

Add custom value

9. In the **Alert Logic** section, select the values for these fields if they are not already automatically selected.

FIELD	DESCRIPTION	VALUE FOR THIS SCENARIO
Operator	The operator to compare the metric value against the threshold	Greater than
Threshold value	The value that the result is measured against.	80
Frequency of evaluation	How often the alert rule should run. A frequency smaller than the aggregation granularity results in a sliding window evaluation.	15 minutes

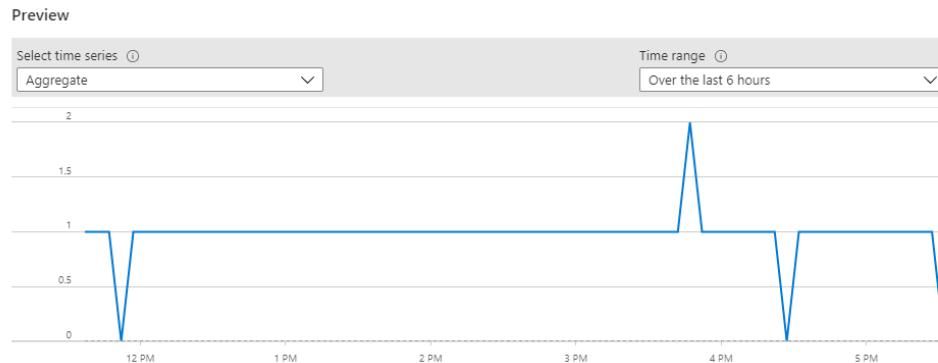
10. (Optional) In the **Advanced options** section, set the **Number of violations to trigger alert**.

Advanced options

Number of violations to trigger the alert:

Number of violations	<input type="text" value="1"/>
Evaluation period	<input type="text" value="5 minutes"/> (1 aggregated points)
Override query time range ⓘ	<input type="text" value="None (5 minutes)"/>

11. The **Preview** chart shows query evaluations results over time. You can change the chart period or select different time series that resulted from unique alert splitting by dimensions.



12. From this point on, you can select the **Review + create** button at any time.

13. In the **Actions** tab, select or create the required **action groups**.

Scope Condition Actions Details Tags Review + create

An action group is a set of actions that can be applied to an alert rule. [Learn more](#)

[+ Add action groups](#) [+ Create action group](#)

Action group name

Contains actions

No action group selected yet

[Review + create](#)

[Previous](#)

[Next: Details >](#)

14. In the **Details** tab, define the **Project details** and the **Alert rule details**.

15. (Optional) In the **Advanced options** section, you can set several options, including whether to **Enable upon creation**, or to **mute actions** for a period after the alert rule fires.

[Create an alert rule](#) ...

Scope Condition Actions **Details** Tags Review + create

#### Project details

Select the subscription and resource group in which to save the alert rule.

Subscription \* ⓘ

Contoso Hotels Tenant - Production

Resource group \* ⓘ

CH1-OpsRG-Pri

[Create new](#)

#### Alert rule details

Severity \* ⓘ

3 - Informational

Alert rule name \* ⓘ

Alert rule description ⓘ

Region \* ⓘ

East US

Advanced options

#### Custom properties

Add your own properties to the alert rule. These will be sent with the alert payload.

Name	Value
	:

#### Settings

Enable upon creation ⓘ



Automatically resolve alerts (preview) ⓘ



Mute actions ⓘ



Check workspace linked storage ⓘ



> [!NOTE] > If you or your administrator assigned the Azure Policy **Azure Log Search Alerts over Log Analytics workspaces should use customer-managed keys**, you must select **Check workspace linked storage** option in **Advanced options**, or the rule creation will fail as it will not meet the policy

requirements.

16. In the **Tags** tab, set any required tags on the alert rule resource.

The screenshot shows the 'Tags' tab of an alert rule configuration. At the top, there are tabs for Scope, Condition, Actions, Details, Tags (which is underlined), and Review + create. Below the tabs, a note states: 'Tags are name and value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. [Learn more about using tags](#)'.

Note that if you later change resource settings on other tabs, your tags will be automatically updated.

Below this, there is a table with two columns: 'Name' and 'Value'. A single row is shown with a placeholder 'Name : Value'. At the bottom of the screen, there are three buttons: 'Review + create' (highlighted in blue), 'Previous', and 'Next: Review + create >'.

17. In the **Review + create** tab, a validation will run and inform you of any issues.

18. When validation passes and you have reviewed the settings, click the **Create** button.

The screenshot shows the 'Review + create' tab of the alert rule configuration. It includes sections for Product Details, Scope, Condition, Alert logic, and Advanced options. The 'Product Details' section shows Log alerts, 1 Condition (highlighted in red), Terms of use | Privacy statement, Total Total pricing (\$), and Pricing. The 'Scope' section shows Resource: ACME-Portal, ACME Telco, and MyResourceGro... The 'Condition' section has a collapsed 'Search query' section. The 'Alert logic' section shows Operator: GreaterThan, Threshold value: 5, and Frequency of evaluation: 5 minutes. The 'Advanced options' section is collapsed. At the bottom, there are 'Create' and 'Previous' buttons.

## Next steps

- [Monitor workloads running on virtual machines.](#)
- [Analyze monitoring data collected for virtual machines.](#)

# Dependency Agent

9/21/2022 • 3 minutes to read • [Edit Online](#)

The Dependency Agent collects data about processes running on the virtual machine and external process dependencies. Dependency Agent updates include bug fixes or support of new features or functionality. This article describes Dependency Agent requirements and how to upgrade Dependency Agent manually or through automation.

## Dependency Agent requirements

- The Dependency Agent requires the Log Analytics Agent to be installed on the same machine.
- On both the Windows and Linux versions, the Dependency Agent collects data using a user-space service and a kernel driver.
  - Dependency Agent supports the same [Windows versions Log Analytics Agent supports](#), except Windows Server 2008 SP2 and Azure Stack HCI.
  - For Linux, see [Dependency Agent Linux support](#).

## Upgrade Dependency Agent

You can upgrade the Dependency agent for Windows and Linux manually or automatically, depending on the deployment scenario and environment the machine is running in, using these methods:

ENVIRONMENT	INSTALLATION METHOD	UPGRADE METHOD
Azure VM	Dependency agent VM extension for <a href="#">Windows</a> and <a href="#">Linux</a>	Agent is automatically upgraded by default unless you configured your Azure Resource Manager template to opt out by setting the property <code>autoUpgradeMinorVersion</code> to <code>false</code> . The upgrade for minor version where auto upgrade is disabled, and a major version upgrade follow the same method - uninstall and reinstall the extension.
Custom Azure VM images	Manual install of Dependency agent for Windows/Linux	Updating VMs to the newest version of the agent needs to be performed from the command line running the Windows installer package or Linux self-extracting and installable shell script bundle.
Non-Azure VMs	Manual install of Dependency agent for Windows/Linux	Updating VMs to the newest version of the agent needs to be performed from the command line running the Windows installer package or Linux self-extracting and installable shell script bundle.

### Upgrade Windows agent

Update the agent on a Windows VM from the command prompt, with a script or other automation solution, or by using the `InstallDependencyAgent-Windows.exe` Setup Wizard.

Download the latest version of the Windows agent.

#### Using the Setup Wizard

1. Sign on to the computer with an account that has administrative rights.
2. Execute `InstallDependencyAgent-Windows.exe` to start the Setup Wizard.
3. Follow the **Dependency Agent Setup** wizard to uninstall the previous version of the dependency agent and then install the latest version.

#### From the command line

1. Sign on to the computer with an account that has administrative rights.
2. Run the following command.

```
InstallDependencyAgent-Windows.exe /S /RebootMode=manual
```

The `/RebootMode=manual` parameter prevents the upgrade from automatically rebooting the machine if some processes are using files from the previous version and have a lock on them.

3. To confirm the upgrade was successful, check the `install.log` for detailed setup information. The log directory is `%Programfiles%\Microsoft Dependency Agent\logs`.

#### Upgrade Linux agent

Upgrade from prior versions of the Dependency agent on Linux is supported and performed following the same command as a new installation.

You can download the latest version of the Linux agent from [here](#).

1. Sign on to the computer with an account that has administrative rights.
2. Run the following command as root.

```
InstallDependencyAgent-Linux64.bin -s
```

If the Dependency agent fails to start, check the logs for detailed error information. On Linux agents, the log directory is `/var/opt/microsoft/dependency-agent/log`.

## Dependency Agent Linux support

Since the Dependency agent works at the kernel level, support is also dependent on the kernel version. As of Dependency agent version 9.10.\* the agent supports \* kernels. The following table lists the major and minor Linux OS release and supported kernel versions for the Dependency agent.

#### NOTE

Dependency agent is not supported for Azure Virtual Machines with Ampere Altra ARM-based processors.

DISTRIBUTION	OS VERSION	KERNEL VERSION
Red Hat Linux 8	8.5	4.18.0-348.*el8_5.x86_64 4.18.0-348.*el8.x86_64
	8.4	4.18.0-305.*el8.x86_64, 4.18.0-305.*el8_4.x86_64

DISTRIBUTION	OS VERSION	KERNEL VERSION
	8.3	4.18.0-240.*el8_3.x86_64
	8.2	4.18.0-193.*el8_2.x86_64
	8.1	4.18.0-147.*el8_1.x86_64
	8.0	4.18.0-80.*el8.x86_64 4.18.0-80.*el8_0.x86_64
Red Hat Linux 7	7.9	3.10.0-1160
	7.8	3.10.0-1136
	7.7	3.10.0-1062
	7.6	3.10.0-957
	7.5	3.10.0-862
	7.4	3.10.0-693
Red Hat Linux 6	6.10	2.6.32-754
	6.9	2.6.32-696
CentOS Linux 8	8.5	4.18.0-348.*el8_5.x86_64 4.18.0-348.*el8.x86_64
	8.4	4.18.0-305.*el8.x86_64, 4.18.0-305.*el8_4.x86_64
	8.3	4.18.0-240.*el8_3.x86_64
	8.2	4.18.0-193.*el8_2.x86_64
	8.1	4.18.0-147.*el8_1.x86_64
	8.0	4.18.0-80.*el8.x86_64 4.18.0-80.*el8_0.x86_64
CentOS Linux 7	7.9	3.10.0-1160
	7.8	3.10.0-1136
	7.7	3.10.0-1062
CentOS Linux 6	6.10	2.6.32-754.3.5 2.6.32-696.30.1
	6.9	2.6.32-696.30.1 2.6.32-696.18.7

DISTRIBUTION	OS VERSION	KERNEL VERSION
Ubuntu Server	20.04	5.8 5.4*
	18.04	5.3.0-1020 5.0 (includes Azure-tuned kernel) 4.18* 4.15*
	16.04.3	4.15.*
	16.04	4.13.* 4.11.* 4.10.* 4.8.* 4.4.*
	14.04	3.13.*-generic 4.4.*-generic
SUSE Linux 12 Enterprise Server	12 SP5	4.12.14-122.*-default, 4.12.14-16.*-azure
	12 SP4	4.12.* (includes Azure-tuned kernel)
	12 SP3	4.4.*
	12 SP2	4.4.*
SUSE Linux 15 Enterprise Server	15 SP1	4.12.14-197.*-default, 4.12.14-8.*-azure
	15	4.12.14-150.*-default
Debian	9	4.9

## Next steps

If you want to stop monitoring your VMs for a while or remove VM insights entirely, see [Disable monitoring of your VMs in VM insights](#).

# Disable monitoring of your VMs in VM insights

9/21/2022 • 3 minutes to read • [Edit Online](#)

After you enable monitoring of your virtual machines (VMs), you can later choose to disable monitoring in VM insights. This article shows how to disable monitoring for one or more VMs.

Currently, VM insights doesn't support selective disabling of VM monitoring. Your Log Analytics workspace might support VM insights and other solutions. It might also collect other monitoring data. If your Log Analytics workspace provides these services, you need to understand the effect and methods of disabling monitoring before you start.

VM insights relies on the following components to deliver its experience:

- A Log Analytics workspace, which stores monitoring data from VMs and other sources.
- A collection of performance counters configured in the workspace. The collection updates the monitoring configuration on all VMs connected to the workspace.
- `VMInsights`, which is a monitoring solution configured in the workspace. This solution updates the monitoring configuration on all VMs connected to the workspace.
- `MicrosoftMonitoringAgent` (for Windows) or `OmsAgentForLinux` (for Linux), and `DependencyAgent`, which are Azure VM extensions. These extensions collect and send data to the workspace.

As you prepare to disable monitoring of your VMs, keep these considerations in mind:

- If you evaluated with a single VM and used the preselected default Log Analytics workspace, you can disable monitoring by uninstalling the Dependency agent from the VM and disconnecting the Log Analytics agent from this workspace. This approach is appropriate if you intend to use the VM for other purposes and decide later to reconnect it to a different workspace.
- If you selected a preexisting Log Analytics workspace that supports other monitoring solutions and data collection from other sources, you can remove solution components from the workspace without interrupting or affecting your workspace.

## NOTE

After removing the solution components from your workspace, you might continue to see performance and map data for your Azure VMs. Data will eventually stop appearing in the **Performance** and **Map** views. The **Enable** option will be available from the selected Azure VM so you can re-enable monitoring in the future.

## Remove VM insights completely

If you still need the Log Analytics workspace, follow these steps to completely remove VM insights. You'll remove the `VMInsights` solution from the workspace.

1. Sign in to the [Azure portal](#).
2. In the Azure portal, select **All services**. In the list of resources, type **Log Analytics**. As you begin typing, the list filters suggestions based on your input. Select **Log Analytics**.
3. In your list of Log Analytics workspaces, select the workspace you chose when you enabled VM insights.
4. On the left, select **Solutions**.
5. In the list of solutions, select **VMInsights(workspace name)**. On the **Overview** page for the solution, select **Delete**. When prompted to confirm, select **Yes**.

## Disable monitoring and keep the workspace

If your Log Analytics workspace still needs to support monitoring from other sources, following these steps to disable monitoring on the VM that you used to evaluate VM insights. For Azure VMs, you'll remove the dependency agent VM extension and the Log Analytics agent VM extension for Windows or Linux directly from the VM.

### NOTE

Don't remove the Log Analytics agent if:

- Azure Automation manages the VM to orchestrate processes or to manage configuration or updates.
- Microsoft Defender for Cloud manages the VM for security and threat detection.

If you do remove the Log Analytics agent, you will prevent those services and solutions from proactively managing your VM.

1. Sign in to the [Azure portal](#).
2. In the Azure portal, select **Virtual Machines**.
3. From the list, select a VM.
4. On the left, select **Extensions**. On the **Extensions** page, select **DependencyAgent**.
5. On the extension properties page, select **Uninstall**.
6. On the **Extensions** page, select **MicrosoftMonitoringAgent**. On the extension properties page, select **Uninstall**.

# Understanding Azure virtual machine usage

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

By analyzing your Azure usage data, powerful consumption insights can be gained – insights that can enable better cost management and allocation throughout your organization. This document provides a deep dive into your Azure Compute consumption details. For more details on general Azure usage, navigate to [Understanding your bill](#).

## Download your usage details

To begin, [download your usage details](#). The table below provides the definition and example values of usage for Virtual Machines deployed via the Azure Resource Manager. This document does not contain detailed information for VMs deployed via our classic model.

FIELD	MEANING	EXAMPLE VALUES
Usage Date	The date when the resource was used	11/23/2017
Meter ID	Identifies the top-level service for which this usage belongs to	Virtual Machines
Meter Sub-Category	<p>The billed meter identifier.</p> <p>For Compute Hour usage, there is a meter for each VM Size + OS (Windows, Non-Windows) + Region.</p> <p>For Premium software usage, there is a meter for each software type. Most premium software images have different meters for each core size. For more information, visit the <a href="#">Compute Pricing Page</a></p>	2005544f-659d-49c9-9094-8e0aea1be3a5
Meter Name	This is specific for each service in Azure. For compute, it is always "Compute Hours".	Compute Hours
Meter Region	Identifies the location of the datacenter for certain services that are priced based on datacenter location.	JA East
Unit	Identifies the unit that the service is charged in. Compute resources are billed per hour.	Hours
Consumed	The amount of the resource that has been consumed for that day. For Compute, we bill for each minute the VM ran for a given hour (up to 6 decimals of accuracy).	1, 0.5
Resource Location	Identifies the datacenter where the resource is running.	JA East
Consumed Service	The Azure platform service that you used.	Microsoft.Compute
Resource Group	The resource group in which the deployed resource is running in. For more information, see <a href="#">Azure Resource Manager overview</a> .	MyRG

FIELD	MEANING	EXAMPLE VALUES
Instance ID	The identifier for the resource. The identifier contains the name you specify for the resource when it was created. For VMs, the Instance ID will contain the SubscriptionId, ResourceGroupName, and VMName (or scale set name for scale set usage).	/subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/MyRG/providers/Microsoft.Compute/virtualMachines/ or /subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/MyRG/providers/Microsoft.Compute/virtualMachines/
Tags	Tag you assign to the resource. Use tags to group billing records. Learn how to tag your Virtual Machines using the <a href="#">CLI</a> or <a href="#">PowerShell</a> . This is available for Resource Manager VMs only.	{"myDepartment": "RD", "myUser": "myName"}
Additional Info	<p>Service-specific metadata. For VMs, we populate the following data in the additional info field:</p> <p>Image Type- specific image that you ran. Find the full list of supported strings below under Image Types.</p> <p>Service Type: the size that you deployed.</p> <p>VMName: name of your VM. This field is only populated for scale set VMs. If you need your VM Name for scale set VMs, you can find that in the Instance ID string above.</p> <p>UsageType: This specifies the type of usage this represents.</p> <p>ComputeHR is the Compute Hour usage for the underlying VM, like Standard_D1_v2.</p> <p>ComputeHR_SW is the premium software charge if the VM is using premium software.</p>	<p>Virtual Machines</p> <pre>{"ImageType": "Canonical", "ServiceType": "Standard_DS1_v2", "VMName": "UsageType": "ComputeHR"}</pre> <p>Virtual Machine Scale Sets</p> <pre>{"ImageType": "Canonical", "ServiceType": "Standard_DS1_v2", "VMName": "UsageType": "ComputeHR"}</pre> <p>Premium Software</p> <pre>{"ImageType": "", "ServiceType": "Standard_DS1_v2", "VMName": "", "UsageType": "ComputeHR_SW"}</pre>

## Image Type

For some images in the Azure gallery, the image type is populated in the Additional Info field. This enables users to understand and track what they have deployed on their Virtual Machine. The following values that are populated in this field based on the image you have deployed:

- BitRock
- Canonical FreeBSD
- Open Logic
- Oracle
- SLES for SAP
- SQL Server 14 Preview on Windows Server 2012 R2 Preview
- SUSE
- SUSE Premium
- StorSimple Cloud Appliance
- Red Hat
- Red Hat for SAP Business Applications
- Red Hat for SAP HANA
- Windows Client BYOL
- Windows Server BYOL
- Windows Server Preview

## Service Type

The service type field in the Additional Info field corresponds to the exact VM size you deployed. Premium storage VMs (SSD-based) and non-premium storage VMs (HDD-based) are priced the same. If you deploy an SSD-based size, like Standard\_DS2\_v2, you see the non-SSD size (`Standard_D2_v2 VM`) in the Meter Sub-Category column and the SSD-size (`Standard_DS2_v2`) in the Additional Info field.

## Region Names

The region name populated in the Resource Location field in the usage details varies from the region name used in the Azure Resource Manager. Here is a mapping between the region values:

RESOURCE MANAGER REGION NAME	RESOURCE LOCATION IN USAGE DETAILS
australiaeast	AU East
australiasoutheast	AU Southeast
brazilsouth	BR South
CanadaCentral	CA Central
CanadaEast	CA East
CentralIndia	IN Central
centralus	Central US
chinaeast	China East
chinanorth	China North
eastasia	East Asia
eastus	East US
eastus2	East US 2
GermanyCentral	DE Central
GermanyNortheast	DE Northeast
japaneast	JA East
japanwest	JA West
KoreaCentral	KR Central
KoreaSouth	KR South
northcentralus	North Central US
northeurope	North Europe
southcentralus	South Central US
southeastasia	Southeast Asia
SouthIndia	IN South
UKNorth	US North
uksouth	UK South
UKSouth2	UK South 2
ukwest	UK West
USDoDCentral	US DoD Central
USDoDEast	US DoD East
USGovArizona	USGov Arizona
usgoviowa	USGov Iowa

RESOURCE MANAGER REGION NAME	RESOURCE LOCATION IN USAGE DETAILS
USGovTexas	USGov Texas
usgovvirginia	USGov Virginia
westcentralus	US West Central
westeurope	West Europe
WestIndia	IN West
westus	West US
westus2	US West 2

## Virtual machine usage FAQ

### What resources are charged when deploying a VM?

VMs acquire costs for the VM itself, any premium software running on the VM, the storage account\managed disk associated with the VM, and the networking bandwidth transfers from the VM.

### How can I tell if a VM is using Azure Hybrid Benefit in the Usage CSV?

If you deploy using the [Azure Hybrid Benefit](#), you are charged the Non-Windows VM rate since you are bringing your own license to the cloud. In your bill, you can distinguish which Resource Manager VMs are running Azure Hybrid Benefit because they have either "Windows\_Server BYOL" or "Windows\_Client BYOL" in the ImageType column.

### How are Basic vs. Standard VM Types differentiated in the Usage CSV?

Both Basic and Standard A-Series VMs are offered. If you deploy a Basic VM, in the Meter Sub Category, it has the string "Basic." If you deploy a Standard A-Series VM, then the VM size appears as "A1 VM" since Standard is the default. To learn more about the differences between Basic and Standard, see the [Pricing Page](#).

### What are ExtraSmall, Small, Medium, Large, and ExtraLarge sizes?

ExtraSmall - ExtraLarge are the legacy names for Standard\_A0 – Standard\_A4. In classic VM usage records, you might see this convention used if you have deployed these sizes.

### What is the difference between Meter Region and Resource Location?

The Meter Region is associated with the meter. For some Azure services who use one price for all regions, the Meter Region field could be blank. However, since VMs have dedicated prices per region for Virtual Machines, this field is populated. Similarly, the Resource Location for Virtual Machines is the location where the VM is deployed. The Azure regions in both fields are the same, although they might have a different string convention for the region name.

### Why is the ImageType value blank in the Additional Info field?

The ImageType field is only populated for a subset of images. If you did not deploy one of the images above, the ImageType is blank.

### Why is the VMName blank in the Additional Info?

The VMName is only populated in the Additional Info field for VMs in a scale set. The InstanceID field contains the VM name for non-scale set VMs.

### What does ComputeHR mean in the UsageType field in the Additional Info?

ComputeHR stands for Compute Hour which represents the usage event for the underlying infrastructure cost. If the UsageType is ComputeHR\_SW, the usage event represents the premium software charge for the VM.

### How do I know if I am charged for premium software?

When exploring which VM Image best fits your needs, be sure to check out [Azure Marketplace](#). The image has the software plan rate. If you see "Free" for the rate, there is no additional cost for the software.

### What is the difference between Microsoft.ClassicCompute and Microsoft.Compute in the Consumed service?

Microsoft.ClassicCompute represents classic resources deployed via the Azure Service Manager. If you deploy via the Resource Manager, then Microsoft.Compute is populated in the consumed service. Learn more about the [Azure Deployment models](#).

### Why is the InstanceID field blank for my Virtual Machine usage?

If you deploy via the classic deployment model, the InstanceID string is not available.

### Why are the tags for my VMs not flowing to the usage details?

Tags flow to the Usage CSV for Resource Manager VMs only. Classic resource tags are not available in the usage details.

**How can the consumed quantity be more than 24 hours one day?**

In the Classic model, billing for resources is aggregated at the Cloud Service level. If you have more than one VM in a Cloud Service that uses the same billing meter, your usage is aggregated together. VMs deployed via Resource Manager are billed at the VM level, so this aggregation will not apply.

**Why is pricing not available for DS/FS/GS/LS sizes on the pricing page?**

Premium storage capable VMs are billed at the same rate as non-premium storage capable VMs. Only your storage costs differ. Visit the [storage pricing page](#) for more information.

## Next steps

To learn more about your usage details, see [Understand your bill for Microsoft Azure](#).

# How to tag a VM using the Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes how to tag a VM using the Azure CLI. Tags are user-defined key/value pairs which can be placed directly on a resource or a resource group. Azure currently supports up to 50 tags per resource and resource group. Tags may be placed on a resource at the time of creation or added to an existing resource. You can also tag a virtual machine using Azure [PowerShell](#).

You can view all properties for a given VM, including the tags, using `az vm show`.

```
az vm show --resource-group myResourceGroup --name myVM --query tags
```

To add a new VM tag through the Azure CLI, you can use the `azure vm update` command along with the tag parameter `--set`:

```
az vm update \
--resource-group myResourceGroup \
--name myVM \
--set tags.myNewTagName1=myNewTagValue1 tags.myNewTagName2=myNewTagValue2
```

To remove tags, you can use the `--remove` parameter in the `azure vm update` command.

```
az vm update \
--resource-group myResourceGroup \
--name myVM \
--remove tags.myNewTagName1
```

Now that we have applied tags to our resources Azure CLI and the Portal, let's take a look at the usage details to see the tags in the billing portal.

## Next steps

- To learn more about tagging your Azure resources, see [Azure Resource Manager Overview](#) and [Using Tags to organize your Azure Resources](#).
- To see how tags can help you manage your use of Azure resources, see [Understanding your Azure Bill](#).

# Tagging a VM using the portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes how to add tags to a VM using the portal. Tags are user-defined key/value pairs which can be placed directly on a resource or a resource group. Azure currently supports up to 50 tags per resource and resource group. Tags may be placed on a resource at the time of creation or added to an existing resource.

1. Navigate to your VM in the portal.
2. In Essentials, select [Click here to add tags](#).

^ Essentials

Resource group ([change](#)) : myResourceGroup

Status : Running

Location : East US 2

Subscription ([change](#)) : myAzureSubscription

Subscription ID : 1010acb-1010-a1b2-c3d4-1010abcd1010

Tags ([change](#)) : [Click here to add tags](#)

3. Add a value for **Name** and **Value**, and then select **Save**.

## Edit tags

X

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups. Tag names are case insensitive, but tag values are case sensitive. [Learn more about tags](#) ↗

### Tags

Name ⓘ	Value ⓘ
<input type="text"/>	: <input type="text"/>

### Resource

 myVM (Virtual machine)
No changes

[Save](#)

[Cancel](#)

## Next steps

- To learn more about tagging your Azure resources, see [Azure Resource Manager Overview](#) and [Using Tags to organize your Azure Resources](#).
- To see how tags can help you manage your use of Azure resources, see [Understanding your Azure Bill](#).

# How to tag a virtual machine in Azure using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes how to tag a VM in Azure using PowerShell. Tags are user-defined key/value pairs which can be placed directly on a resource or a resource group. Azure currently supports up to 50 tags per resource and resource group. Tags may be placed on a resource at the time of creation or added to an existing resource. If you want to tag a virtual machine using the Azure CLI, see [How to tag a virtual machine in Azure using the Azure CLI](#).

Use the `Get-AzVM` cmdlet to view the current list of tags for your VM.

```
Get-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM" | Format-List -Property Tags
```

If your Virtual Machine already contains tags, you will then see all the tags in list format.

To add tags, use the `Set-AzResource` command. When updating tags through PowerShell, tags are updated as a whole. If you are adding one tag to a resource that already has tags, you will need to include all the tags that you want to be placed on the resource. Below is an example of how to add additional tags to a resource through PowerShell Cmdlets.

Assign all of the current tags for the VM to the `$tags` variable, using the `Get-AzResource` and `Tags` property.

```
$tags = (Get-AzResource -ResourceGroupName myResourceGroup -Name myVM).Tags
```

To see the current tags, type the variable.

```
$tags
```

Here is what the output might look like:

Key	Value
Department	MyDepartment
Application	MyApp1
Created By	MyName
Environment	Production

In the following example, we add a tag called `Location` with the value `myLocation`. Use `+=` to append the new key/value pair to the `$tags` list.

```
$tags += @{Location="myLocation"}
```

Use `Set-AzResource` to set all of the tags defined in the `$tags` variable on the VM.

```
Set-AzResource -ResourceGroupName myResourceGroup -Name myVM -ResourceType "Microsoft.Compute/VirtualMachines" -Tag $tags
```

Use `Get-AzResource` to display all of the tags on the resource.

```
(Get-AzResource -ResourceGroupName myResourceGroup -Name myVM).Tags
```

The output should look something like the following, which now includes the new tag:

Key	Value
----	-----
Department	MyDepartment
Application	MyApp1
Created By	MyName
Environment	Production
Location	MyLocation

## Next steps

- To learn more about tagging your Azure resources, see [Azure Resource Manager Overview](#) and [Using Tags to organize your Azure Resources](#).
- To see how tags can help you manage your use of Azure resources, see [Understanding your Azure Bill](#).

# Tagging a VM using a template

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

This article describes how to tag a VM in Azure using a Resource Manager template. Tags are user-defined key/value pairs which can be placed directly on a resource or a resource group. Azure currently supports up to 50 tags per resource and resource group. Tags may be placed on a resource at the time of creation or added to an existing resource.

This template places tags on the following resources: Compute (Virtual Machine), Storage (Storage Account), and Network (Public IP Address, Virtual Network, and Network Interface). This template is for a Windows VM but can be adapted for Linux VMs.

Click the **Deploy to Azure** button from the [template link](#). This will navigate to the [Azure portal](#) where you can deploy this template.

## Simple deployment of a VM with Tags

 Deploy to Azure

 Visualize

This template includes the following tags: *Department*, *Application*, and *Created By*. You can add/edit these tags directly in the template if you would like different tag names.

```
"apiVersion": "2015-05-01-preview",
"type": "Microsoft.Compute/virtualMachines",
"name": "[variables('vmName')]",
"location": "[variables('location')]",
"tags": {
    "Department": "[parameters('departmentName')]",
    "Application": "[parameters('applicationName')]",
    "Created By": "[parameters('createdBy')]"
},
```

As you can see, the tags are defined as key/value pairs, separated by a colon (:). The tags must be defined in this format:

```
"tags": {
    "Key1" : "Value1",
    "Key2" : "Value2"
}
```

Save the template file after you finish editing it with the tags of your choice.

Next, in the **Edit Parameters** section, you can fill out the values for your tags.

DEPARTMENTNAME (string) ⓘ
MyDepartment
APPLICATIONNAME (string) ⓘ
MyApp
CREATEDBY (string) ⓘ
MyName

Click **Create** to deploy this template with your tag values.

### Next steps

- To learn more about tagging your Azure resources, see [Azure Resource Manager Overview](#) and [Using Tags to organize your Azure Resources](#).
- To see how tags can help you manage your use of Azure resources, see [Understanding your Azure Bill](#).

# Azure Instance Metadata Service (Linux)

9/21/2022 • 36 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

The Azure Instance Metadata Service (IMDS) provides information about currently running virtual machine instances. You can use it to manage and configure your virtual machines. This information includes the SKU, storage, network configurations, and upcoming maintenance events. For a complete list of the data available, see the [Endpoint Categories Summary](#).

IMDS is available for running instances of virtual machines (VMs) and virtual machine scale set instances. All endpoints support VMs created and managed by using [Azure Resource Manager](#). Only the Attested category and Network portion of the Instance category support VMs created by using the classic deployment model. The Attested endpoint does so only to a limited extent.

IMDS is a REST API that's available at a well-known, non-routable IP address (`169.254.169.254`). You can only access it from within the VM. Communication between the VM and IMDS never leaves the host. Have your HTTP clients bypass web proxies within the VM when querying IMDS, and treat `169.254.169.254` the same as `168.63.129.16`.

## Usage

### Access Azure Instance Metadata Service

To access IMDS, create a VM from [Azure Resource Manager](#) or the [Azure portal](#), and use the following samples. For more examples, see [Azure Instance Metadata Samples](#).

Here's sample code to retrieve all metadata for an instance. To access a specific data source, see [Endpoint Categories](#) for an overview of all available features.

### Request

#### IMPORTANT

This example bypasses proxies. You **must** bypass proxies when querying IMDS. See [Proxies](#) for additional information.

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance?api-version=2021-02-01" | ConvertTo-Json -Depth 64
```

`-NoProxy` requires PowerShell V6 or greater. See our [samples repository](#) for examples with older PowerShell versions.

### Response

#### NOTE

The response is a JSON string. The following example response is pretty-printed for readability.

- [Windows](#)
- [Linux](#)

```
{
  "compute": {
    "azEnvironment": "AZUREPUBLICCLOUD",
    "additionalCapabilities": {
      "hibernationEnabled": "true"
    },
    "hostGroup": {
      "id": "testHostGroupId"
    },
    "extendedLocation": {
      "type": "edgeZone",
      "name": "microsoftlosangeles"
    },
    "evictionPolicy": "",
    "isHostCompatibilityLayerVm": "true",
    "licenseType": "Windows_Client",
    "location": "westus",
    "name": "examplevmname",
    "offer": "WindowsServer",
    "osProfile": {
      "adminUsername": "admin",
      "computerName": "examplevmname",
      "disablePasswordAuthentication": "true"
    },
    "osType": "Windows",
    "placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
    "plan": {
      "name": "planName",
      "product": "planProduct",
      "publisher": "planPublisher"
    },
    "platformFaultDomain": "36",
    "platformSubFaultDomain": "",
    "platformUpdateDomain": "42",
    "priority": "Regular",
    "publicKeys": [
      {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
      },
      {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
      }
    ],
    "publisher": "RDFE-Test-Microsoft-Windows-Server-Group",
    "resourceGroupName": "macikgo-test-may-23",
    "resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
    "securityProfile": {
      "secureBootEnabled": "true",
      "virtualTpmEnabled": "false",
      "encryptionAtHost": "true",
      "securityType": "TrustedLaunch"
    },
    "sku": "2019-Datacenter",
    "storageProfile": {
      "dataDisks": [
        {
          "bytesPerSecondThrottle": "979202048",
          "caching": "None",
          "createOption": "Empty",
          "diskCapacityBytes": "274877906944",
          "diskSizeGB": "1024",
          "image": {
            "uri": ""
          }
        }
      ]
    }
  }
}
```

```
        },
        "isSharedDisk": "false",
        "isUltraDisk": "true",
        "lun": "0",
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampledatadiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampledatadiskname",
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    ],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
```

```

    "subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
    "tags": "baz:bash;foo:bar",
    "userData": "Zm9vYmFy",
    "version": "15.05.22",
    "virtualMachineScaleSet": {
        "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
    },
    "vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
    "vmScaleSetName": "crpteste9vflji9",
    "vmSize": "Standard_A3",
    "zone": ""
},
"network": {
    "interface": [
        "ipv4": {
            "ipAddress": [
                "privateIpAddress": "10.144.133.132",
                "publicIpAddress": ""
            ],
            "subnet": [
                {
                    "address": "10.144.133.128",
                    "prefix": "26"
                }
            ],
            "ipv6": {
                "ipAddress": [
                ]
            },
            "macAddress": "0011AAFFBB22"
        }
    ]
}
}

```

## Security and authentication

The Instance Metadata Service is only accessible from within a running virtual machine instance on a non-routable IP address. VMs are limited to interacting with metadata/functionality that pertains to themselves. The API is HTTP only and never leaves the host.

In order to ensure that requests are directly intended for IMDS and prevent unintended or unwanted redirection of requests, requests:

- Must contain the header `Metadata: true`
- Must **not** contain an `X-Forwarded-For` header

Any request that does not meet **both** of these requirements will be rejected by the service.

### IMPORTANT

IMDS is **not** a channel for sensitive data. The API is unauthenticated and open to all processes on the VM. Information exposed through this service should be considered as shared information to all applications running inside the VM.

If it is not necessary for every process on the VM to access IMDS endpoint, you can set local firewall rules to limit the access. For example, if only a known system service needs to access instance metadata service, you can set a firewall rule on IMDS endpoint, only allowing the specific process(es) to access, or denying access for the rest of the processes.

## Proxies

IMDS is **not** intended to be used behind a proxy and doing so is unsupported. Most HTTP clients provide an

option for you to disable proxies on your requests, and this functionality must be utilized when communicating with IMDS. Consult your client's documentation for details.

#### IMPORTANT

Even if you don't know of any proxy configuration in your environment, **you still must override any default client proxy settings**. Proxy configurations can be automatically discovered, and failing to bypass such configurations exposes you to outage risks should the machine's configuration be changed in the future.

## Rate limiting

In general, requests to IMDS are limited to 5 requests per second (on a per VM basis). Requests exceeding this threshold will be rejected with 429 responses. Requests to the [Managed Identity](#) category are limited to 20 requests per second and 5 concurrent requests.

## HTTP verbs

The following HTTP verbs are currently supported:

VERB	DESCRIPTION
GET	Retrieve the requested resource

## Parameters

Endpoints may support required and/or optional parameters. See [Schema](#) and the documentation for the specific endpoint in question for details.

### Query parameters

IMDS endpoints support HTTP query string parameters. For example:

```
http://169.254.169.254/metadata/instance/compute?api-version=2021-01-01&format=json
```

Specifies the parameters:

NAME	VALUE
api-version	2021-01-01
format	json

Requests with duplicate query parameter names will be rejected.

### Route parameters

For some endpoints that return larger json blobs, we support appending route parameters to the request endpoint to filter down to a subset of the response:

```
http://169.254.169.254/metadata/<endpoint>/[<filter parameter>/...]?<query parameters>
```

The parameters correspond to the indexes/keys that would be used to walk down the json object were you interacting with a parsed representation.

For example, `/metatadata/instance` returns the json object:

```
{  
    "compute": { ... },  
    "network": {  
        "interface": [  
            {  
                "ipv4": {  
                    "ipAddress": [{  
                        "privateIpAddress": "10.144.133.132",  
                        "publicIpAddress": ""  
                    }],  
                    "subnet": [{  
                        "address": "10.144.133.128",  
                        "prefix": "26"  
                    }]  
                },  
                "ipv6": {  
                    "ipAddress": [  
                        ...  
                    ]  
                },  
                "macAddress": "0011AAFFBB22"  
            },  
            ...  
        ]  
    }  
}
```

If we want to filter the response down to just the compute property, we would send the request:

```
http://169.254.169.254/metadata/instance/compute?api-version=<version>
```

Similarly, if we want to filter to a nested property or specific array element we keep appending keys:

```
http://169.254.169.254/metadata/instance/network/interface/0?api-version=<version>
```

would filter to the first element from the `Network.interface` property and return:

```
{  
    "ipv4": {  
        "ipAddress": [{  
            "privateIpAddress": "10.144.133.132",  
            "publicIpAddress": ""  
        }],  
        "subnet": [{  
            "address": "10.144.133.128",  
            "prefix": "26"  
        }]  
    },  
    "ipv6": {  
        "ipAddress": [  
            ...  
        ]  
    },  
    "macAddress": "0011AAFFBB22"  
}
```

#### NOTE

When filtering to a leaf node, `format=json` doesn't work. For these queries `format=text` needs to be explicitly specified since the default format is json.

## Schema

### Data format

By default, IMDS returns data in JSON format (`Content-Type: application/json`). However, endpoints that support response filtering (see [Route Parameters](#)) also support the format `text`.

To access a non-default response format, specify the requested format as a query string parameter in the request. For example:

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance?api-version=2017-08-01&format=text"
```

In json responses, all primitives will be of type `string`, and missing or inapplicable values are always included but will be set to an empty string.

### Versioning

IMDS is versioned and specifying the API version in the HTTP request is mandatory. The only exception to this requirement is the [versions](#) endpoint, which can be used to dynamically retrieve the available API versions.

As newer versions are added, older versions can still be accessed for compatibility if your scripts have dependencies on specific data formats.

When you don't specify a version, you get an error with a list of the newest supported versions:

```
{
    "error": "Bad request. api-version was not specified in the request. For more information refer to aka.ms/azureimds",
    "newest-versions": [
        "2020-10-01",
        "2020-09-01",
        "2020-07-15"
    ]
}
```

### Supported API versions

- 2017-03-01
- 2017-04-02
- 2017-08-01
- 2017-10-01
- 2017-12-01
- 2018-02-01
- 2018-04-02
- 2018-10-01
- 2019-02-01

- 2019-03-11
- 2019-04-30
- 2019-06-01
- 2019-06-04
- 2019-08-01
- 2019-08-15
- 2019-11-01
- 2020-06-01
- 2020-07-15
- 2020-09-01
- 2020-10-01
- 2020-12-01
- 2021-01-01
- 2021-02-01
- 2021-03-01
- 2021-05-01
- 2021-10-01

## Swagger

A full Swagger definition for IMDS is available at: <https://github.com/Azure/azure-rest-api-specs/blob/main/specification/imds/data-plane/readme.md>

## Regional availability

The service is **generally available** in all Azure Clouds.

## Root endpoint

The root endpoint is `http://169.254.169.254/metadata`.

## Endpoint categories

The IMDS API contains multiple endpoint categories representing different data sources, each of which contains one or more endpoints. See each category for details.

CATEGORY ROOT	DESCRIPTION	VERSION INTRODUCED
<code>/metadata/attested</code>	See <a href="#">Attested Data</a>	2018-10-01
<code>/metadata/identity</code>	See <a href="#">Managed Identity via IMDS</a>	2018-02-01
<code>/metadata/instance</code>	See <a href="#">Instance Metadata</a>	2017-04-02
<code>/metadata/loadbalancer</code>	See <a href="#">Retrieve Load Balancer metadata via IMDS</a>	2020-10-01
<code>/metadata/scheduledevents</code>	See <a href="#">Scheduled Events via IMDS</a>	2017-08-01
<code>/metadata/versions</code>	See <a href="#">Versions</a>	N/A

# Versions

## NOTE

This feature was released alongside version 2020-10-01, which is currently being rolled out and may not yet be available in every region.

## List API versions

Returns the set of supported API versions.

```
GET /metadata/versions
```

### Parameters

None (this endpoint is unversioned).

### Response

```
{
  "apiVersions": [
    "2017-03-01",
    "2017-04-02",
    ...
  ]
}
```

# Instance metadata

## Get VM metadata

Exposes the important metadata for the VM instance, including compute, network, and storage.

```
GET /metadata/instance
```

### Parameters

NAME	REQUIRED/OPTIONAL	DESCRIPTION
<code>api-version</code>	Required	The version used to service the request.
<code>format</code>	Optional*	The format ( <code>json</code> or <code>text</code> ) of the response. *Note: May be required when using request parameters

This endpoint supports response filtering via [route parameters](#).

### Response

- [Windows](#)
- [Linux](#)

```
{
  "compute": {
    "azEnvironment": "AZUREPUBLICCLOUD",
    "additionalCapabilities": {
      "hibernationEnabled": "true"
    }
  }
}
```

```
},
"hostGroup": {
    "id": "testHostGroupId"
},
"extendedLocation": {
    "type": "edgeZone",
    "name": "microsoftlosangeles"
},
"evictionPolicy": "",
"isHostCompatibilityLayerVm": "true",
"licenseType": "Windows_Client",
"location": "westus",
"name": "examplevmname",
"offer": "WindowsServer",
"osProfile": {
    "adminUsername": "admin",
    "computerName": "examplevmname",
    "disablePasswordAuthentication": "true"
},
"osType": "Windows",
"placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
"plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
},
"platformFaultDomain": "36",
"platformSubFaultDomain": "",
"platformUpdateDomain": "42",
"priority": "Regular",
"publicKeys": [
    {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
    },
    {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
    }
],
"publisher": "RDFE-Test-Microsoft-Windows-Server-Group",
"resourceGroupName": "macikgo-test-may-23",
"resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
"securityProfile": {
    "secureBootEnabled": "true",
    "virtualTpmEnabled": "false",
    "encryptionAtHost": "true",
    "securityType": "TrustedLaunch"
},
"sku": "2019-Datacenter",
"storageProfile": {
    "dataDisks": [
        {
            "bytesPerSecondThrottle": "979202048",
            "caching": "None",
            "createOption": "Empty",
            "diskCapacityBytes": "274877906944",
            "diskSizeGB": "1024",
            "image": {
                "uri": ""
            },
            "isSharedDisk": "false",
            "isUltraDisk": "true",
            "lun": "0",
            "managedDisk": {
                "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampledatadiskname",
                "storageAccountType": "StandardSSD_LRS"
            },
            "name": "exampledatadiskname",
            "osType": "Windows"
        }
    ]
}
```

```
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    ],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/macikgo-test-may-
23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
"subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
"tags": "baz:bash;foo:bar",
"userData": "Zm9vYmFy",
"version": "15.05.22",
"virtualMachineScaleSet": {
    "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-
name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
},
"vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
"vmScaleSetName": "crpteste9vflji9",
"vmSize": "Standard_A3",
```

```

        "zone": "",
    },
    "network": {
        "interface": [
            "ipv4": {
                "ipAddress": [
                    {
                        "privateIpAddress": "10.144.133.132",
                        "publicIpAddress": ""
                    }
                ],
                "subnet": [
                    {
                        "address": "10.144.133.128",
                        "prefix": "26"
                    }
                ]
            },
            "ipv6": {
                "ipAddress": [
                ]
            },
            "macAddress": "0011AAFFBB22"
        ]
    }
}
}

```

Schema breakdown:

## Compute

DATA	DESCRIPTION	VERSION INTRODUCED
<code>azEnvironment</code>	Azure Environment where the VM is running in	2018-10-01
<code>additionalCapabilities.hibernationEnabled</code>	Identifies if hibernation is enabled on the VM	2021-11-01+
<code>customData</code>	This feature is deprecated and disabled <a href="#">in IMDS</a> . It has been superseded by <code>userData</code>	2019-02-01
<code>evictionPolicy</code>	Sets how a <a href="#">Spot VM</a> will be evicted.	2020-12-01
<code>extendedLocation.type</code>	Type of the extended location of the VM.	2021-03-01
<code>extendedLocation.name</code>	Name of the extended location of the VM	2021-03-01
<code>host.id</code>	Name of the host of the VM. Note that a VM will either have a host or a hostGroup but not both.	2021-11-15+
<code>hostGroup.id</code>	Name of the hostGroup of the VM. Note that a VM will either have a host or a hostGroup but not both.	2021-11-15+
<code>isHostCompatibilityLayerVm</code>	Identifies if the VM runs on the Host Compatibility Layer	2020-06-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>licenseType</code>	Type of license for <a href="#">Azure Hybrid Benefit</a> . This is only present for AHB-enabled VMs	2020-09-01
<code>location</code>	Azure Region the VM is running in	2017-04-02
<code>name</code>	Name of the VM	2017-04-02
<code>offer</code>	Offer information for the VM image and is only present for images deployed from Azure image gallery	2017-04-02
<code>osProfile.adminUsername</code>	Specifies the name of the admin account	2020-07-15
<code>osProfile.computerName</code>	Specifies the name of the computer	2020-07-15
<code>osProfile.disablePasswordAuthentication</code>	Specifies if password authentication is disabled. This is only present for Linux VMs	2020-10-01
<code>osType</code>	Linux or Windows	2017-04-02
<code>placementGroupId</code>	<a href="#">Placement Group</a> of your virtual machine scale set	2017-08-01
<code>plan</code>	<a href="#">Plan</a> containing name, product, and publisher for a VM if it is an Azure Marketplace Image	2018-04-02
<code>platformUpdateDomain</code>	<a href="#">Update domain</a> the VM is running in	2017-04-02
<code>platformFaultDomain</code>	<a href="#">Fault domain</a> the VM is running in	2017-04-02
<code>platformSubFaultDomain</code>	Sub fault domain the VM is running in, if applicable.	2021-10-01
<code>priority</code>	Priority of the VM. Refer to <a href="#">Spot VMs</a> for more information	2020-12-01
<code>provider</code>	Provider of the VM	2018-10-01
<code>publicKeys</code>	<a href="#">Collection of Public Keys</a> assigned to the VM and paths	2018-04-02
<code>publisher</code>	Publisher of the VM image	2017-04-02
<code>resourceGroupName</code>	<a href="#">Resource group</a> for your Virtual Machine	2017-08-01
<code>resourceId</code>	The <a href="#">fully qualified</a> ID of the resource	2019-03-11

DATA	DESCRIPTION	VERSION INTRODUCED
<code>sku</code>	Specific SKU for the VM image	2017-04-02
<code>securityProfile.secureBootEnabled</code>	Identifies if UEFI secure boot is enabled on the VM	2020-06-01
<code>securityProfile.virtualTpmEnabled</code>	Identifies if the virtual Trusted Platform Module (TPM) is enabled on the VM	2020-06-01
<code>securityProfile.encryptionAtHost</code>	Identifies if <a href="#">Encryption at Host</a> is enabled on the VM	2021-11-01†
<code>securityProfile.securityType</code>	Identifies if the VM is a <a href="#">Trusted VM</a> or a <a href="#">Confidential VM</a>	2021-12-13†
<code>storageProfile</code>	See Storage Profile below	2019-06-01
<code>subscriptionId</code>	Azure subscription for the Virtual Machine	2017-08-01
<code>tags</code>	<a href="#">Tags</a> for your Virtual Machine	2017-08-01
<code>tagsList</code>	Tags formatted as a JSON array for easier programmatic parsing	2019-06-04
<code>userData</code>	The set of data specified when the VM was created for use during or after provisioning (Base64 encoded)	2021-01-01
<code>version</code>	Version of the VM image	2017-04-02
<code>virtualMachineScaleSet.id</code>	ID of the <a href="#">Virtual Machine Scale Set created with flexible orchestration</a> the Virtual Machine is part of. This field is not available for Virtual Machine Scale Sets created with uniform orchestration.	2021-03-01
<code>vmId</code>	<a href="#">Unique identifier</a> for the VM. The blog referenced only suits for VMs that have SMBIOS < 2.6. For VMs that have SMBIOS >= 2.6, the UUID from DMI is displayed in little-endian format, thus, there is no requirement to switch bytes.	2017-04-02
<code>vmScaleSetName</code>	<a href="#">Virtual machine scale set Name</a> of your virtual machine scale set	2017-12-01
<code>vmSize</code>	<a href="#">VM size</a>	2017-04-02
<code>zone</code>	<a href="#">Availability Zone</a> of your virtual machine	2017-12-01

† This version is not fully available yet and may not be supported in all regions.

## Storage profile

The storage profile of a VM is divided into three categories: image reference, OS disk, and data disks, plus an additional object for the local temporary disk.

The image reference object contains the following information about the OS image:

DATA	DESCRIPTION
<code>id</code>	Resource ID
<code>offer</code>	Offer of the platform or marketplace image
<code>publisher</code>	Image publisher
<code>sku</code>	Image sku
<code>version</code>	Version of the platform or marketplace image

The OS disk object contains the following information about the OS disk used by the VM:

DATA	DESCRIPTION
<code>caching</code>	Caching requirements
<code>createOption</code>	Information about how the VM was created
<code>diffDiskSettings</code>	Ephemeral disk settings
<code>diskSizeGB</code>	Size of the disk in GB
<code>image</code>	Source user image virtual hard disk
<code>managedDisk</code>	Managed disk parameters
<code>name</code>	Disk name
<code>vhd</code>	Virtual hard disk
<code>writeAcceleratorEnabled</code>	Whether or not writeAccelerator is enabled on the disk

The data disks array contains a list of data disks attached to the VM. Each data disk object contains the following information:

DATA	DESCRIPTION	VERSION INTRODUCED
<code>bytesPerSecondThrottle</code> *	Disk read/write quota in bytes	2021-05-01
<code>caching</code>	Caching requirements	2019-06-01
<code>createOption</code>	Information about how the VM was created	2019-06-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>diffDiskSettings</code>	Ephemeral disk settings	2019-06-01
<code>diskCapacityBytes</code> *	Size of disk in bytes	2021-05-01
<code>diskSizeGB</code>	Size of the disk in GB	2019-06-01
<code>encryptionSettings</code>	Encryption settings for the disk	2019-06-01
<code>image</code>	Source user image virtual hard disk	2019-06-01
<code>isSharedDisk</code> *	Identifies if the disk is shared between resources	2021-05-01
<code>isUltraDisk</code>	Identifies if the data disk is an Ultra Disk	2021-05-01
<code>lun</code>	Logical unit number of the disk	2019-06-01
<code>managedDisk</code>	Managed disk parameters	2019-06-01
<code>name</code>	Disk name	2019-06-01
<code>opsPerSecondThrottle</code> *	Disk read/write quota in IOPS	2021-05-01
<code>osType</code>	Type of OS included in the disk	2019-06-01
<code>vhd</code>	Virtual hard disk	2019-06-01
<code>writeAcceleratorEnabled</code>	Whether or not writeAccelerator is enabled on the disk	2019-06-01

\* These fields are only populated for Ultra Disks; they will be empty strings from non-Ultra Disks.

The encryption settings blob contains data about how the disk is encrypted (if it is encrypted):

DATA	DESCRIPTION	VERSION INTRODUCED
<code>diskEncryptionKey.sourceVault.id</code>	The location of the disk encryption key	2021-11-01†
<code>diskEncryptionKey.secretUrl</code>	The location of the secret	2021-11-01†
<code>keyEncryptionKey.sourceVault.id</code>	The location of the key encryption key	2021-11-01†
<code>keyEncryptionKey.keyUrl</code>	The location of the key	2021-11-01†

\† This version is not fully available yet and may not be supported in all regions.

The resource disk object contains the size of the [Local Temp Disk](#) attached to the VM, if it has one, in kilobytes. If there is [no local temp disk for the VM](#), this value is 0.

DATA	DESCRIPTION	VERSION INTRODUCED
<code>resourceDisk.size</code>	Size of the local temp disk for the VM (in kB)	2021-02-01

## Network

DATA	DESCRIPTION	VERSION INTRODUCED
<code>ipv4.privateIpAddress</code>	Local IPv4 address of the VM	2017-04-02
<code>ipv4.publicIpAddress</code>	Public IPv4 address of the VM	2017-04-02
<code>subnet.address</code>	Subnet address of the VM	2017-04-02
<code>subnet.prefix</code>	Subnet prefix, example 24	2017-04-02
<code>ipv6.ipAddress</code>	Local IPv6 address of the VM	2017-04-02
<code>macAddress</code>	VM mac address	2017-04-02

### NOTE

The nics returned by the network call are not guaranteed to be in order.

## Get user data

When creating a new VM, you can specify a set of data to be used during or after the VM provision, and retrieve it through IMDS. Check the end to end user data experience [here](#).

To set up user data, utilize the quickstart template [here](#). The sample below shows how to retrieve this data through IMDS. This feature is released with version `2021-01-01` and above.

### NOTE

Security notice: IMDS is open to all applications on the VM, sensitive data should not be placed in the user data.

- [Windows](#)
- [Linux](#)

```
$userData = Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance/compute/userData?api-version=2021-01-01&format=text" [System.Text.Encoding]::UTF8.GetString([Convert]::FromBase64String($userData))
```

### Sample 1: Tracking VM running on Azure

As a service provider, you may require to track the number of VMs running your software or have agents that need to track uniqueness of the VM. To be able to get a unique ID for a VM, use the `vmId` field from Instance Metadata Service.

## Request

- [Windows](#)

- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/vmId?api-version=2017-08-01&format=text"
```

## Response

```
5c08b38e-4d57-4c23-ac45-aca61037f084
```

### Sample 2: Placement of different data replicas

For certain scenarios, placement of different data replicas is of prime importance. For example, [HDFS replica placement](#) or container placement via an [orchestrator](#) might require you to know the `platformFaultDomain` and `platformUpdateDomain` the VM is running on. You can also use [Availability Zones](#) for the instances to make these decisions. You can query this data directly via IMDS.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/platformFaultDomain?api-version=2017-08-01&format=text"
```

## Response

```
0
```

### Sample 3: Get VM tags

VM tags are included the instance API under `instance/compute/tags` endpoint. Tags may have been applied to your Azure VM to logically organize them into a taxonomy. The tags assigned to a VM can be retrieved by using the request below.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/tags?api-version=2017-08-01&format=text"
```

## Response

```
Department:IT;ReferenceNumber:123456;TestStatus:Pending
```

The `tags` field is a string with the tags delimited by semicolons. This output can be a problem if semicolons are used in the tags themselves. If a parser is written to programmatically extract the tags, you should rely on the `tagsList` field. The `tagsList` field is a JSON array with no delimiters, and consequently, easier to parse. The `tagsList` assigned to a VM can be retrieved by using the request below.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance/compute/tagsList?api-version=2019-06-04" | ConvertTo-Json -Depth 64
```

## Response

- [Windows](#)
- [Linux](#)

```
{
  "value": [
    {
      "name": "Department",
      "value": "IT"
    },
    {
      "name": "ReferenceNumber",
      "value": "123456"
    },
    {
      "name": "TestStatus",
      "value": "Pending"
    }
  ],
  "Count": 3
}
```

## Sample 4: Get more information about the VM during support case

As a service provider, you may get a support call where you would like to know more information about the VM. Asking the customer to share the compute metadata can provide basic information for the support professional to know about the kind of VM on Azure.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance/compute?api-version=2020-09-01" | ConvertTo-Json -Depth 64
```

## Response

### NOTE

The response is a JSON string. The following example response is pretty-printed for readability.

- [Windows](#)
- [Linux](#)

```
{
  "azEnvironment": "AZUREPUBLICCLOUD",
```

```
"extendedLocation": {
    "type": "edgeZone",
    "name": "microsoftlosangeles"
},
"evictionPolicy": "",
"additionalCapabilities": {
    "hibernationEnabled": "false"
},
"hostGroup": {
    "id": "testHostGroupId"
},
"isHostCompatibilityLayerVm": "true",
"licenseType": "Windows_Client",
"location": "westus",
"name": "examplevmname",
"offer": "WindowsServer",
"osProfile": {
    "adminUsername": "admin",
    "computerName": "examplevmname",
    "disablePasswordAuthentication": "true"
},
"osType": "Windows",
"placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
"plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
},
"platformFaultDomain": "36",
"platformUpdateDomain": "42",
"priority": "Regular",
"publicKeys": [
    {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
    },
    {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
    }
],
"publisher": "RDDE-Test-Microsoft-Windows-Server-Group",
"resourceGroupName": "macikgo-test-may-23",
"resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
"securityProfile": {
    "secureBootEnabled": "true",
    "virtualTpmEnabled": "false",
    "encryptionAtHost": "true",
    "securityType": "TrustedLaunch"
},
"sku": "2019-Datacenter",
"storageProfile": {
    "dataDisks": [
        {
            "bytesPerSecondThrottle": "979202048",
            "caching": "None",
            "createOption": "Empty",
            "diskCapacityBytes": "274877906944",
            "diskSizeGB": "1024",
            "image": {
                "uri": ""
            },
            "isSharedDisk": "false",
            "isUltraDisk": "true",
            "lun": "0",
            "managedDisk": {
                "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/MicrosoftCompute/disks/exampledatadiskname",
                "storageAccountType": "StandardSSD_LRS"
            }
        }
    ]
}
```

```
        "name": "exampleddatadiskname",
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    }],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxx-
xxxx-xxxx-xxxx-xxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-
23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
"subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
"tags": "baz:bash;foo:bar",
"version": "15.05.22",
"virtualMachineScaleSet": {
    "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-
name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
},
"vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
"vmScaleSetName": "crpteste9vflji9",
"vmSize": "Standard A3".
```

```
        "zone": ""  
    }  
}
```

#### Sample 5: Get the Azure Environment where the VM is running

Azure has various sovereign clouds like [Azure Government](#). Sometimes you need the Azure Environment to make some runtime decisions. The following sample shows you how you can achieve this behavior.

##### Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/azEnvironment?api-version=2018-10-01&format=text"
```

##### Response

```
AzurePublicCloud
```

The cloud and the values of the Azure environment are listed here.

CLOUD	AZURE ENVIRONMENT
All generally available global Azure regions	AzurePublicCloud
Azure Government	AzureUSGovernmentCloud
Azure China 21Vianet	AzureChinaCloud
Azure Germany	AzureGermanCloud

#### Sample 6: Retrieve network information

##### Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/network?api-version=2017-08-01" | ConvertTo-Json -Depth 64
```

##### Response

```
{
  "interface": [
    {
      "ipv4": {
        "ipAddress": [
          {
            "privateIpAddress": "10.1.0.4",
            "publicIpAddress": "X.X.X.X"
          }
        ],
        "subnet": [
          {
            "address": "10.1.0.0",
            "prefix": "24"
          }
        ]
      },
      "ipv6": {
        "ipAddress": []
      },
      "macAddress": "000D3AF806EC"
    }
  ]
}
```

#### Sample 7: Retrieve public IP address

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
"http://169.254.169.254/metadata/instance/network/interface/0/ipv4/ipAddress/0/publicIpAddress?api-
version=2017-08-01&format=text"
```

#### NOTE

- If you are looking to retrieve IMDS information for **Standard** SKU Public IP address, review [Load Balancer Metadata API](#) for more information.

## Attested data

### Get Attested data

IMDS helps to provide guarantees that the data provided is coming from Azure. Microsoft signs part of this information, so you can confirm that an image in Azure Marketplace is the one you are running on Azure.

```
GET /metadata/attested/document
```

#### Parameters

NAME	REQUIRED/OPTIONAL	DESCRIPTION
<code>api-version</code>	Required	The version used to service the request.

Name	Required/Optional	Description
nonce	Optional	A 10-digit string that serves as a cryptographic nonce. If no value is provided, IMDS uses the current UTC timestamp.

## Response

```
{
  "encoding": "pkcs7",

  "signature": "MIIEEgYJKoZIhvcNAQcCoIIEAzCCA/8CAQExDzANBgkqhkiG9w0BAQsFADCBugYJKoZIhvcNAQcBoIGsBIGpeyJub25jZSI
6IjEyMzQ1NjY3NjYiLCJwbGFuIjp7Im5hbWUiOiIiLCJwcm9kdWN0IjoiiIwicHVibGlzaGVyIjoiIn0sInRpblVtDGFtcCI6eyJjcmVhdGV
kt24i0IiXMS8yMC8xOCAYmjowNzozOSAtMDAwMCIsImV4cGlyZXNPbiI6IjExLzIwLzE4IDIyOjA40jI0IC0wMDAwIn0sInZtSWQiOiiifaC
CAj8wggI7MIIbPkadAgEcAhBnxW5Kh8ds1EBA0E2mIBj0MA0GCSqGSIB3DQEBAUAMCsxKTAAnBGNVBAMTIHrlc3RzdWJkb21haW4ubWv0YWR
hdGEuYXp1cmUuY29tMB4XDTE4MTEyMDIxNTc1N1oXDTE4MTIyMDIxNTc1NlowKzEpMcC GA1UEAxMgdGVzdHN1YmRvbWFpbis5tZRhZGF0YS5
henVz5jb20wgZ8wDQYJKoZIhvcNAQEBQAdgY0AMIGJAoGBAML/tBo86ENWPzmXZ0kPkX5dY5QZ150ma8lommse71x2sCLonzv4/Ulk4H
+jMMWRRwIea2CuQ5RhdWAhvKq6if4okKnt66fxm+YTVz9z0CTfCmLT+nsdf0AsG1xZppEapC0Cd9vD6NCKyE8aYI1pliaeOnFjG0WvMY04u
Wz2MdAgMBAAGjYDBeMFwGA1UdAQRVMFOAEnYkHLa04Ut4Mpt7TkJFFyhLTArMSkwJwYDVQQDEyB0ZXN0c3ViZG9tYwluLm1ldGFkYXRhLmF
6dXJ1LmNvbYIQZ8VuSofHbJRAQNBnpiAsdDANBgkqhkiG9w0BAQFAAOBgQCLSM6aX5Bs1KHCrp4VQtzPzXF71rVKCocHy3N9PTJQ9Fpnd+
bYw2vSpQHg/AiG82WuDFpPreJvr7Pa938mZqW9HUOGjQKK2FYDTg6FXD8pkPdygh1X5boGWAMMrf7bFkup+1sT+n2tRw2wbNkn01tQ0wICq
y2VqzWwLi45RBwTGB6DCB5QIBATA/MCsxtAnBGNVBAMTIHrlc3RzdWJkb21haW4ubWv0YWRhdGEuYXp1cmUuY29tAhBnxW5Kh8ds1EBA0E2
mIBj0MA0GCSqGSIB3DQEBCwUAMA0GCSqGSIB3DQEBAQUABIGAl1d1BM/yYIqqv8SDE4kjQo3U1/IKAVR8ETKcve5BAdGSNkTUooUGVniTxev
Dj5NkmazOaKZp9fEtByqqPOyw/n1xaZg0044HDG1PUJ90xVYmfek6p9RpJBu6kiKhnnYTelUk5u75phe5ZbMFbhupPhXmYAdjc7Nmw97nx8N
nprQ="
}
```

The signature blob is a [pkcs7](#)-signed version of document. It contains the certificate used for signing along with certain VM-specific details.

For VMs created by using Azure Resource Manager, the document includes `vmId`, `sku`, `nonce`, `subscriptionId`, `timeStamp` for creation and expiry of the document, and the plan information about the image. The plan information is only populated for Azure Marketplace images.

For VMs created by using the classic deployment model, only the `vmId` and `subscriptionId` are guaranteed to be populated. You can extract the certificate from the response, and use it to confirm that the response is valid and is coming from Azure.

The decoded document contains the following fields:

Data	Description	Version Introduced
<code>licenseType</code>	Type of license for <a href="#">Azure Hybrid Benefit</a> . This is only present for AHB-enabled VMs.	2020-09-01
<code>nonce</code>	A string that can be optionally provided with the request. If no <code>nonce</code> was supplied, the current Coordinated Universal Time timestamp is used.	2018-10-01
<code>plan</code>	The <a href="#">Azure Marketplace Image plan</a> . Contains the plan ID (name), product image or offer (product), and publisher ID (publisher).	2018-10-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>timestamp.createdOn</code>	The UTC timestamp for when the signed document was created	2018-20-01
<code>timestamp.expiresOn</code>	The UTC timestamp for when the signed document expires	2018-10-01
<code>vmId</code>	Unique identifier for the VM	2018-10-01
<code>subscriptionId</code>	Azure subscription for the Virtual Machine	2019-04-30
<code>sku</code>	Specific SKU for the VM image (correlates to <code>compute/sku</code> property from the Instance Metadata endpoint [ <code>/metadata/instance</code> ])	2019-11-01

#### NOTE

For Classic (non-Azure Resource Manager) VMs, only the `vmId` is guaranteed to be populated.

Example document:

```
{
  "nonce": "20201130-211924",
  "plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
  },
  "sku": "Windows-Server-2012-R2-Datacenter",
  "subscriptionId": "8d10da13-8125-4ba9-a717-bf7490507b3d",
  "timeStamp": {
    "createdOn": "11/30/20 21:19:19 -0000",
    "expiresOn": "11/30/20 21:19:24 -0000"
  },
  "vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6"
}
```

#### Sample 1: Validate that the VM is running in Azure

Vendors in Azure Marketplace want to ensure that their software is licensed to run only in Azure. If someone copies the VHD to an on-premises environment, the vendor needs to be able to detect that. Through IMDS, these vendors can get signed data that guarantees response only from Azure.

#### NOTE

This sample requires the jq utility to be installed.

## Validation

- [Windows](#)
- [Linux](#)

```

# Get the signature
$attestedDoc = Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
http://169.254.169.254/metadata/attested/document?api-version=2020-09-01
# Decode the signature
$signature = [System.Convert]::FromBase64String($attestedDoc.signature)

```

Verify that the signature is from Microsoft Azure and check the certificate chain for errors.

```

# Get certificate chain
$cert = [System.Security.Cryptography.X509Certificates.X509Certificate2]($signature)
$chain = New-Object -TypeName System.Security.Cryptography.X509Certificates.X509Chain
$chain.Build($cert)
# Print the Subject of each certificate in the chain
foreach($element in $chain.ChainElements)
{
    Write-Host $element.Certificate.Subject
}

# Get the content of the signed document
Add-Type -AssemblyName System.Security
$signedCms = New-Object -TypeName System.Security.Cryptography.Pkcs.SignedCms
$signedCms.Decode($signature);
$content = [System.Text.Encoding]::UTF8.GetString($signedCms.ContentInfo.Content)
Write-Host "Attested data: " $content
$json = $content | ConvertFrom-Json
# Do additional validation here

```

#### NOTE

Due to IMDS's caching mechanism, a previously cached `nonce` value might be returned.

The `nonce` in the signed document can be compared if you provided a `nonce` parameter in the initial request.

#### NOTE

The certificate for the public cloud and each sovereign cloud will be different.

CLOUD	CERTIFICATE
All generally available global Azure regions	*.metadata.azure.com
Azure Government	*.metadata.azure.us
Azure China 21Vianet	*.metadata.azure.cn
Azure Germany	*.metadata.microsoftazure.de

#### NOTE

The certificates might not have an exact match of `metadata.azure.com` for the public cloud. For this reason, the certification validation should allow a common name from any `.metadata.azure.com` subdomain.

In cases where the intermediate certificate can't be downloaded due to network constraints during validation, you can pin the intermediate certificate. Azure rolls over the certificates, which is standard PKI practice. You must

update the pinned certificates when rollover happens. Whenever a change to update the intermediate certificate is planned, the Azure blog is updated, and Azure customers are notified.

You can find the intermediate certificates on [this page](#). The intermediate certificates for each of the regions can be different.

#### NOTE

The intermediate certificate for Azure China 21Vianet will be from DigiCert Global Root CA, instead of Baltimore. If you pinned the intermediate certificates for Azure China as part of a root chain authority change, the intermediate certificates must be updated.

#### NOTE

Starting February 2022, our Attested Data certificates will be impacted by a TLS change. Due to this, the root CA will change from Baltimore CyberTrust to DigiCert Global G2 only for Public and US Government clouds. If you have the Baltimore CyberTrust cert or other intermediate certificates listed in [this post](#) pinned, please follow the instructions listed there **immediately** to prevent any disruptions from using the Attested Data endpoint.

## Managed identity

A managed identity, assigned by the system, can be enabled on the VM. You can also assign one or more user-assigned managed identities to the VM. You can then request tokens for managed identities from IMDS. Use these tokens to authenticate with other Azure services, such as Azure Key Vault.

For detailed steps to enable this feature, see [Acquire an access token](#).

## Load Balancer Metadata

When you place virtual machine or virtual machine set instances behind an Azure Standard Load Balancer, you can use IMDS to retrieve metadata related to the load balancer and the instances. For more information, see [Retrieve load balancer information](#).

## Scheduled events

You can obtain the status of the scheduled events by using IMDS. Then the user can specify a set of actions to run upon these events. For more information, see [Scheduled events for Linux](#) or [Scheduled events for Windows](#).

## Sample code in different languages

The following table lists samples of calling IMDS by using different languages inside the VM:

LANGUAGE	EXAMPLE
Bash	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.sh">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.sh</a>
C#	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.cs">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.cs</a>
Go	<a href="https://github.com/Microsoft/azureimds/blob/master/imdssample.go">https://github.com/Microsoft/azureimds/blob/master/imdssample.go</a>

LANGUAGE	EXAMPLE
Java	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.java">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.java</a>
NodeJS	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.js">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.js</a>
Perl	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.pl">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.pl</a>
PowerShell	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.ps1">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.ps1</a>
Puppet	<a href="https://github.com/keirans/azurometadata">https://github.com/keirans/azurometadata</a>
Python	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.py">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.py</a>
Ruby	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.rb">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.rb</a>

## Errors and debugging

If there is a data element not found or a malformed request, the Instance Metadata Service returns standard HTTP errors. For example:

HTTP STATUS CODE	REASON
200 OK	The request was successful.
400 Bad Request	Missing <code>Metadata: true</code> header or missing parameter <code>format=json</code> when querying a leaf node
404 Not Found	The requested element doesn't exist
405 Method Not Allowed	The HTTP method (verb) is not supported on the endpoint.
410 Gone	Retry after some time for a max of 70 seconds
429 Too Many Requests	API <a href="#">Rate Limits</a> have been exceeded
500 Service Error	Retry after some time

## Frequently asked questions

- I am getting the error `400 Bad Request, Required metadata header not specified`. What does this mean?
  - IMDS requires the header `Metadata: true` to be passed in the request. Passing this header in the REST call allows access to IMDS.
- Why am I not getting compute information for my VM?
  - Currently, IMDS only supports instances created with Azure Resource Manager.

- I created my VM through Azure Resource Manager some time ago. Why am I not seeing compute metadata information?
  - If you created your VM after September 2016, add a [tag](#) to start seeing compute metadata. If you created your VM before September 2016, add or remove extensions or data disks to the VM instance to refresh metadata.
- Is user data the same as custom data?
  - User data offers the similar functionality to custom data, allowing you to pass your own metadata to the VM instance. The difference is, user data is retrieved through IMDS, and is persistent throughout the lifetime of the VM instance. Existing custom data feature will continue to work as described in [this article](#). However you can only get custom data through local system folder, not through IMDS.
- Why am I not seeing all data populated for a new version?
  - If you created your VM after September 2016, add a [tag](#) to start seeing compute metadata. If you created your VM before September 2016, add or remove extensions or data disks to the VM instance to refresh metadata.
- Why am I getting the error `500 Internal Server Error` or `410 Resource Gone ?`
  - Retry your request. For more information, see [Transient fault handling](#). If the problem persists, create a support issue in the Azure portal for the VM.
- Would this work for virtual machine scale set instances?
  - Yes, IMDS is available for virtual machine scale set instances.
- I updated my tags in virtual machine scale sets, but they don't appear in the instances (unlike single instance VMs). Am I doing something wrong?
  - Currently tags for virtual machine scale sets only show to the VM on a reboot, reimagine, or disk change to the instance.
- Why am I am not seeing the SKU information for my VM in `instance/compute` details?
  - For custom images created from Azure Marketplace, Azure platform doesn't retain the SKU information for the custom image and the details for any VMs created from the custom image. This is by design and hence not surfaced in the VM `instance/compute` details.
- Why is my request timed out for my call to the service?
  - Metadata calls must be made from the primary IP address assigned to the primary network card of the VM. Additionally, if you've changed your routes, there must be a route for the 169.254.169.254/32 address in your VM's local routing table.
    - [Windows](#)
    - [Linux](#)

1. Dump your local routing table and look for the IMDS entry. For example:

```

> route print
IPv4 Route Table
=====
Active Routes:
Network Destination      Netmask        Gateway       Interface Metric
          0.0.0.0        0.0.0.0    172.16.69.1   172.16.69.7    10
         127.0.0.0    255.0.0.0     On-link        127.0.0.1    331
         127.0.0.1  255.255.255.255     On-link        127.0.0.1    331
  127.255.255.255  255.255.255.255     On-link        127.0.0.1    331
  168.63.129.16  255.255.255.255    172.16.69.1   172.16.69.7    11
 169.254.169.254  255.255.255.255    172.16.69.1   172.16.69.7    11
... (continues) ...

```

2. Verify that a route exists for `169.254.169.254`, and note the corresponding network interface (for example, `172.16.69.7`).
3. Dump the interface configuration and find the interface that corresponds to the one referenced in the routing table, noting the MAC (physical) address.

```

> ipconfig /all
... (continues) ...
Ethernet adapter Ethernet:

Connection-specific DNS Suffix . : xic3mnxjiefupcwr1mcs1rjqiqa.cx.internal.cloudapp.net
Description . . . . . : Microsoft Hyper-V Network Adapter
Physical Address. . . . . : 00-0D-3A-E5-1C-C0
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . : fe80::3166:ce5a:2bd5:a6d1%3(Preferred)
IPv4 Address. . . . . : 172.16.69.7(Preferred)
Subnet Mask . . . . . : 255.255.255.0
... (continues) ...

```

4. Confirm that the interface corresponds to the VM's primary NIC and primary IP. You can find the primary NIC and IP by looking at the network configuration in the Azure portal, or by looking it up with the Azure CLI. Note the private IPs (and the MAC address if you're using the CLI). Here's a PowerShell CLI example:

```

$ResourceGroup = '<Resource_Group>'
$VmName = '<VM_Name>'
$NicNames = az vm nic list --resource-group $ResourceGroup --vm-name $VmName |
ConvertFrom-Json | Foreach-Object { $_.id.Split('/')[-1] }
foreach($NicName in $NicNames)
{
    $Nic = az vm nic show --resource-group $ResourceGroup --vm-name $VmName --nic
    $NicName | ConvertFrom-Json
    Write-Host $NicName, $Nic.primary, $Nic.macAddress
}
# Output: wintest767 True 00-0D-3A-E5-1C-C0

```

5. If they don't match, update the routing table so that the primary NIC and IP are targeted.

- Failover clustering in Windows Server
  - When you're querying IMDS with failover clustering, it's sometimes necessary to add a route to the routing table. Here's how:
    1. Open a command prompt with administrator privileges.
    2. Run the following command, and note the address of the Interface for Network Destination (`0.0.0.0`) in the IPv4 Route Table.

```
route print
```

#### NOTE

The following example output is from a Windows Server VM with failover cluster enabled. For simplicity, the output contains only the IPv4 Route Table.

#### IPv4 Route Table

Active Routes:					
Network Destination	Netmask	Gateway	Interface	Metric	
0.0.0.0	0.0.0.0	10.0.1.1	10.0.1.10	266	
10.0.1.0	255.255.255.192	On-link	10.0.1.10	266	
10.0.1.10	255.255.255.255	On-link	10.0.1.10	266	
10.0.1.15	255.255.255.255	On-link	10.0.1.10	266	
10.0.1.63	255.255.255.255	On-link	10.0.1.10	266	
127.0.0.0	255.0.0.0	On-link	127.0.0.1	331	
127.0.0.1	255.255.255.255	On-link	127.0.0.1	331	
127.255.255.255	255.255.255.255	On-link	127.0.0.1	331	
169.254.0.0	255.255.0.0	On-link	169.254.1.156	271	
169.254.1.156	255.255.255.255	On-link	169.254.1.156	271	
169.254.255.255	255.255.255.255	On-link	169.254.1.156	271	
224.0.0.0	240.0.0.0	On-link	127.0.0.1	331	
224.0.0.0	240.0.0.0	On-link	169.254.1.156	271	
255.255.255.255	255.255.255.255	On-link	127.0.0.1	331	
255.255.255.255	255.255.255.255	On-link	169.254.1.156	271	
255.255.255.255	255.255.255.255	On-link	10.0.1.10	266	

Run the following command and use the address of the Interface for Network Destination ( `0.0.0.0` ), which is ( `10.0.1.10` ) in this example.

```
route add 169.254.169.254/32 10.0.1.10 metric 1 -p
```

## Support

If you aren't able to get a metadata response after multiple attempts, you can create a support issue in the Azure portal.

## Product feedback

You can provide product feedback and ideas to our user feedback channel under Virtual Machines > Instance Metadata Service [here](#)

## Next steps

- [Acquire an access token for the VM](#)
- [Scheduled events for Linux](#)
- [Scheduled events for Windows](#)

# Azure Instance Metadata Service (Windows)

9/21/2022 • 36 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

The Azure Instance Metadata Service (IMDS) provides information about currently running virtual machine instances. You can use it to manage and configure your virtual machines. This information includes the SKU, storage, network configurations, and upcoming maintenance events. For a complete list of the data available, see the [Endpoint Categories Summary](#).

IMDS is available for running instances of virtual machines (VMs) and virtual machine scale set instances. All endpoints support VMs created and managed by using [Azure Resource Manager](#). Only the Attested category and Network portion of the Instance category support VMs created by using the classic deployment model. The Attested endpoint does so only to a limited extent.

IMDS is a REST API that's available at a well-known, non-routable IP address (`169.254.169.254`). You can only access it from within the VM. Communication between the VM and IMDS never leaves the host. Have your HTTP clients bypass web proxies within the VM when querying IMDS, and treat `169.254.169.254` the same as `168.63.129.16`.

## Usage

### Access Azure Instance Metadata Service

To access IMDS, create a VM from [Azure Resource Manager](#) or the [Azure portal](#), and use the following samples. For more examples, see [Azure Instance Metadata Samples](#).

Here's sample code to retrieve all metadata for an instance. To access a specific data source, see [Endpoint Categories](#) for an overview of all available features.

### Request

#### IMPORTANT

This example bypasses proxies. You **must** bypass proxies when querying IMDS. See [Proxies](#) for additional information.

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance?api-version=2021-02-01" | ConvertTo-Json -Depth 64
```

`-NoProxy` requires PowerShell V6 or greater. See our [samples repository](#) for examples with older PowerShell versions.

### Response

#### NOTE

The response is a JSON string. The following example response is pretty-printed for readability.

- [Windows](#)
- [Linux](#)

```
{
  "compute": {
    "azEnvironment": "AZUREPUBLICCLOUD",
    "additionalCapabilities": {
      "hibernationEnabled": "true"
    },
    "hostGroup": {
      "id": "testHostGroupId"
    },
    "extendedLocation": {
      "type": "edgeZone",
      "name": "microsoftlosangeles"
    },
    "evictionPolicy": "",
    "isHostCompatibilityLayerVm": "true",
    "licenseType": "Windows_Client",
    "location": "westus",
    "name": "examplevmname",
    "offer": "WindowsServer",
    "osProfile": {
      "adminUsername": "admin",
      "computerName": "examplevmname",
      "disablePasswordAuthentication": "true"
    },
    "osType": "Windows",
    "placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
    "plan": {
      "name": "planName",
      "product": "planProduct",
      "publisher": "planPublisher"
    },
    "platformFaultDomain": "36",
    "platformSubFaultDomain": "",
    "platformUpdateDomain": "42",
    "priority": "Regular",
    "publicKeys": [
      {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
      },
      {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
      }
    ],
    "publisher": "RDFE-Test-Microsoft-Windows-Server-Group",
    "resourceGroupName": "macikgo-test-may-23",
    "resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
    "securityProfile": {
      "secureBootEnabled": "true",
      "virtualTpmEnabled": "false",
      "encryptionAtHost": "true",
      "securityType": "TrustedLaunch"
    },
    "sku": "2019-Datacenter",
    "storageProfile": {
      "dataDisks": [
        {
          "bytesPerSecondThrottle": "979202048",
          "caching": "None",
          "createOption": "Empty",
          "diskCapacityBytes": "274877906944",
          "diskSizeGB": "1024",
          "image": {
            "uri": ""
          }
        }
      ]
    }
  }
}
```

```
        },
        "isSharedDisk": "false",
        "isUltraDisk": "true",
        "lun": "0",
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampledatadiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampledatadiskname",
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    ],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
```

```

    "subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
    "tags": "baz:bash;foo:bar",
    "userData": "Zm9vYmFy",
    "version": "15.05.22",
    "virtualMachineScaleSet": {
        "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
    },
    "vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
    "vmScaleSetName": "crpteste9vflji9",
    "vmSize": "Standard_A3",
    "zone": ""
},
"network": {
    "interface": [
        "ipv4": {
            "ipAddress": [
                "privateIpAddress": "10.144.133.132",
                "publicIpAddress": ""
            ],
            "subnet": [
                {
                    "address": "10.144.133.128",
                    "prefix": "26"
                }
            ],
            "macAddress": "0011AAFFBB22"
        },
        "ipv6": {
            "ipAddress": [
                ""
            ]
        }
    ],
    "macAddress": "0011AAFFBB22"
}
}
}

```

## Security and authentication

The Instance Metadata Service is only accessible from within a running virtual machine instance on a non-routable IP address. VMs are limited to interacting with metadata/functionality that pertains to themselves. The API is HTTP only and never leaves the host.

In order to ensure that requests are directly intended for IMDS and prevent unintended or unwanted redirection of requests, requests:

- Must contain the header `Metadata: true`
- Must not contain an `X-Forwarded-For` header

Any request that does not meet **both** of these requirements will be rejected by the service.

### IMPORTANT

IMDS is **not** a channel for sensitive data. The API is unauthenticated and open to all processes on the VM. Information exposed through this service should be considered as shared information to all applications running inside the VM.

If it is not necessary for every process on the VM to access IMDS endpoint, you can set local firewall rules to limit the access. For example, if only a known system service needs to access instance metadata service, you can set a firewall rule on IMDS endpoint, only allowing the specific process(es) to access, or denying access for the rest of the processes.

## Proxies

IMDS is **not** intended to be used behind a proxy and doing so is unsupported. Most HTTP clients provide an

option for you to disable proxies on your requests, and this functionality must be utilized when communicating with IMDS. Consult your client's documentation for details.

#### IMPORTANT

Even if you don't know of any proxy configuration in your environment, **you still must override any default client proxy settings**. Proxy configurations can be automatically discovered, and failing to bypass such configurations exposes you to outage risks should the machine's configuration be changed in the future.

## Rate limiting

In general, requests to IMDS are limited to 5 requests per second (on a per VM basis). Requests exceeding this threshold will be rejected with 429 responses. Requests to the [Managed Identity](#) category are limited to 20 requests per second and 5 concurrent requests.

## HTTP verbs

The following HTTP verbs are currently supported:

VERB	DESCRIPTION
GET	Retrieve the requested resource

## Parameters

Endpoints may support required and/or optional parameters. See [Schema](#) and the documentation for the specific endpoint in question for details.

### Query parameters

IMDS endpoints support HTTP query string parameters. For example:

```
http://169.254.169.254/metadata/instance/compute?api-version=2021-01-01&format=json
```

Specifies the parameters:

NAME	VALUE
api-version	2021-01-01
format	json

Requests with duplicate query parameter names will be rejected.

### Route parameters

For some endpoints that return larger json blobs, we support appending route parameters to the request endpoint to filter down to a subset of the response:

```
http://169.254.169.254/metadata/<endpoint>/[<filter parameter>/...]?<query parameters>
```

The parameters correspond to the indexes/keys that would be used to walk down the json object were you interacting with a parsed representation.

For example, `/metatadata/instance` returns the json object:

```
{  
    "compute": { ... },  
    "network": {  
        "interface": [  
            {  
                "ipv4": {  
                    "ipAddress": [{  
                        "privateIpAddress": "10.144.133.132",  
                        "publicIpAddress": ""  
                    }],  
                    "subnet": [{  
                        "address": "10.144.133.128",  
                        "prefix": "26"  
                    }]  
                },  
                "ipv6": {  
                    "ipAddress": [  
                        ...  
                    ]  
                },  
                "macAddress": "0011AAFFBB22"  
            },  
            ...  
        ]  
    }  
}
```

If we want to filter the response down to just the compute property, we would send the request:

```
http://169.254.169.254/metadata/instance/compute?api-version=<version>
```

Similarly, if we want to filter to a nested property or specific array element we keep appending keys:

```
http://169.254.169.254/metadata/instance/network/interface/0?api-version=<version>
```

would filter to the first element from the `Network.interface` property and return:

```
{  
    "ipv4": {  
        "ipAddress": [{  
            "privateIpAddress": "10.144.133.132",  
            "publicIpAddress": ""  
        }],  
        "subnet": [{  
            "address": "10.144.133.128",  
            "prefix": "26"  
        }]  
    },  
    "ipv6": {  
        "ipAddress": [  
            ...  
        ]  
    },  
    "macAddress": "0011AAFFBB22"  
}
```

#### NOTE

When filtering to a leaf node, `format=json` doesn't work. For these queries `format=text` needs to be explicitly specified since the default format is json.

## Schema

### Data format

By default, IMDS returns data in JSON format (`Content-Type: application/json`). However, endpoints that support response filtering (see [Route Parameters](#)) also support the format `text`.

To access a non-default response format, specify the requested format as a query string parameter in the request. For example:

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance?api-version=2017-08-01&format=text"
```

In json responses, all primitives will be of type `string`, and missing or inapplicable values are always included but will be set to an empty string.

### Versioning

IMDS is versioned and specifying the API version in the HTTP request is mandatory. The only exception to this requirement is the [versions](#) endpoint, which can be used to dynamically retrieve the available API versions.

As newer versions are added, older versions can still be accessed for compatibility if your scripts have dependencies on specific data formats.

When you don't specify a version, you get an error with a list of the newest supported versions:

```
{
    "error": "Bad request. api-version was not specified in the request. For more information refer to aka.ms/azureimds",
    "newest-versions": [
        "2020-10-01",
        "2020-09-01",
        "2020-07-15"
    ]
}
```

### Supported API versions

- 2017-03-01
- 2017-04-02
- 2017-08-01
- 2017-10-01
- 2017-12-01
- 2018-02-01
- 2018-04-02
- 2018-10-01
- 2019-02-01

- 2019-03-11
- 2019-04-30
- 2019-06-01
- 2019-06-04
- 2019-08-01
- 2019-08-15
- 2019-11-01
- 2020-06-01
- 2020-07-15
- 2020-09-01
- 2020-10-01
- 2020-12-01
- 2021-01-01
- 2021-02-01
- 2021-03-01
- 2021-05-01
- 2021-10-01

## Swagger

A full Swagger definition for IMDS is available at: <https://github.com/Azure/azure-rest-api-specs/blob/main/specification/imds/data-plane/readme.md>

## Regional availability

The service is **generally available** in all Azure Clouds.

## Root endpoint

The root endpoint is `http://169.254.169.254/metadata`.

## Endpoint categories

The IMDS API contains multiple endpoint categories representing different data sources, each of which contains one or more endpoints. See each category for details.

CATEGORY ROOT	DESCRIPTION	VERSION INTRODUCED
<code>/metadata/attested</code>	See <a href="#">Attested Data</a>	2018-10-01
<code>/metadata/identity</code>	See <a href="#">Managed Identity via IMDS</a>	2018-02-01
<code>/metadata/instance</code>	See <a href="#">Instance Metadata</a>	2017-04-02
<code>/metadata/loadbalancer</code>	See <a href="#">Retrieve Load Balancer metadata via IMDS</a>	2020-10-01
<code>/metadata/scheduledevents</code>	See <a href="#">Scheduled Events via IMDS</a>	2017-08-01
<code>/metadata/versions</code>	See <a href="#">Versions</a>	N/A

# Versions

## NOTE

This feature was released alongside version 2020-10-01, which is currently being rolled out and may not yet be available in every region.

## List API versions

Returns the set of supported API versions.

```
GET /metadata/versions
```

### Parameters

None (this endpoint is unversioned).

### Response

```
{
  "apiVersions": [
    "2017-03-01",
    "2017-04-02",
    ...
  ]
}
```

# Instance metadata

## Get VM metadata

Exposes the important metadata for the VM instance, including compute, network, and storage.

```
GET /metadata/instance
```

### Parameters

NAME	REQUIRED/OPTIONAL	DESCRIPTION
<code>api-version</code>	Required	The version used to service the request.
<code>format</code>	Optional*	The format ( <code>json</code> or <code>text</code> ) of the response. *Note: May be required when using request parameters

This endpoint supports response filtering via [route parameters](#).

### Response

- [Windows](#)
- [Linux](#)

```
{
  "compute": {
    "azEnvironment": "AZUREPUBLICCLOUD",
    "additionalCapabilities": {
      "hibernationEnabled": "true"
    }
  }
}
```

```
},
"hostGroup": {
    "id": "testHostGroupId"
},
"extendedLocation": {
    "type": "edgeZone",
    "name": "microsoftlosangeles"
},
"evictionPolicy": "",
"isHostCompatibilityLayerVm": "true",
"licenseType": "Windows_Client",
"location": "westus",
"name": "examplevmname",
"offer": "WindowsServer",
"osProfile": {
    "adminUsername": "admin",
    "computerName": "examplevmname",
    "disablePasswordAuthentication": "true"
},
"osType": "Windows",
"placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
"plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
},
"platformFaultDomain": "36",
"platformSubFaultDomain": "",
"platformUpdateDomain": "42",
"priority": "Regular",
"publicKeys": [
    {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
    },
    {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
    }
],
"publisher": "RDFE-Test-Microsoft-Windows-Server-Group",
"resourceGroupName": "macikgo-test-may-23",
"resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
"securityProfile": {
    "secureBootEnabled": "true",
    "virtualTpmEnabled": "false",
    "encryptionAtHost": "true",
    "securityType": "TrustedLaunch"
},
"sku": "2019-Datacenter",
"storageProfile": {
    "dataDisks": [
        {
            "bytesPerSecondThrottle": "979202048",
            "caching": "None",
            "createOption": "Empty",
            "diskCapacityBytes": "274877906944",
            "diskSizeGB": "1024",
            "image": {
                "uri": ""
            },
            "isSharedDisk": "false",
            "isUltraDisk": "true",
            "lun": "0",
            "managedDisk": {
                "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/disks/exampledatadiskname",
                "storageAccountType": "StandardSSD_LRS"
            },
            "name": "exampledatadiskname",
            "osType": "Windows"
        }
    ]
}
```

```
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    ],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourceGroups/macikgo-test-may-
23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
"subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
"tags": "baz:bash;foo:bar",
"userData": "Zm9vYmFy",
"version": "15.05.22",
"virtualMachineScaleSet": {
    "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-
name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
},
"vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
"vmScaleSetName": "crpteste9vflji9",
"vmSize": "Standard_A3",
```

```

        "zone": "",
    },
    "network": {
        "interface": [
            "ipv4": {
                "ipAddress": [
                    {
                        "privateIpAddress": "10.144.133.132",
                        "publicIpAddress": ""
                    }
                ],
                "subnet": [
                    {
                        "address": "10.144.133.128",
                        "prefix": "26"
                    }
                ]
            },
            "ipv6": {
                "ipAddress": [
                ]
            },
            "macAddress": "0011AAFFBB22"
        ]
    }
}
}

```

Schema breakdown:

## Compute

DATA	DESCRIPTION	VERSION INTRODUCED
<code>azEnvironment</code>	Azure Environment where the VM is running in	2018-10-01
<code>additionalCapabilities.hibernationEnabled</code>	Identifies if hibernation is enabled on the VM	2021-11-01+
<code>customData</code>	This feature is deprecated and disabled <a href="#">in IMDS</a> . It has been superseded by <code>userData</code>	2019-02-01
<code>evictionPolicy</code>	Sets how a <a href="#">Spot VM</a> will be evicted.	2020-12-01
<code>extendedLocation.type</code>	Type of the extended location of the VM.	2021-03-01
<code>extendedLocation.name</code>	Name of the extended location of the VM	2021-03-01
<code>host.id</code>	Name of the host of the VM. Note that a VM will either have a host or a hostGroup but not both.	2021-11-15+
<code>hostGroup.id</code>	Name of the hostGroup of the VM. Note that a VM will either have a host or a hostGroup but not both.	2021-11-15+
<code>isHostCompatibilityLayerVm</code>	Identifies if the VM runs on the Host Compatibility Layer	2020-06-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>licenseType</code>	Type of license for <a href="#">Azure Hybrid Benefit</a> . This is only present for AHB-enabled VMs	2020-09-01
<code>location</code>	Azure Region the VM is running in	2017-04-02
<code>name</code>	Name of the VM	2017-04-02
<code>offer</code>	Offer information for the VM image and is only present for images deployed from Azure image gallery	2017-04-02
<code>osProfile.adminUsername</code>	Specifies the name of the admin account	2020-07-15
<code>osProfile.computerName</code>	Specifies the name of the computer	2020-07-15
<code>osProfile.disablePasswordAuthentication</code>	Specifies if password authentication is disabled. This is only present for Linux VMs	2020-10-01
<code>osType</code>	Linux or Windows	2017-04-02
<code>placementGroupId</code>	<a href="#">Placement Group</a> of your virtual machine scale set	2017-08-01
<code>plan</code>	<a href="#">Plan</a> containing name, product, and publisher for a VM if it is an Azure Marketplace Image	2018-04-02
<code>platformUpdateDomain</code>	<a href="#">Update domain</a> the VM is running in	2017-04-02
<code>platformFaultDomain</code>	<a href="#">Fault domain</a> the VM is running in	2017-04-02
<code>platformSubFaultDomain</code>	Sub fault domain the VM is running in, if applicable.	2021-10-01
<code>priority</code>	Priority of the VM. Refer to <a href="#">Spot VMs</a> for more information	2020-12-01
<code>provider</code>	Provider of the VM	2018-10-01
<code>publicKeys</code>	<a href="#">Collection of Public Keys</a> assigned to the VM and paths	2018-04-02
<code>publisher</code>	Publisher of the VM image	2017-04-02
<code>resourceGroupName</code>	<a href="#">Resource group</a> for your Virtual Machine	2017-08-01
<code>resourceId</code>	The <a href="#">fully qualified</a> ID of the resource	2019-03-11

DATA	DESCRIPTION	VERSION INTRODUCED
<code>sku</code>	Specific SKU for the VM image	2017-04-02
<code>securityProfile.secureBootEnabled</code>	Identifies if UEFI secure boot is enabled on the VM	2020-06-01
<code>securityProfile.virtualTpmEnabled</code>	Identifies if the virtual Trusted Platform Module (TPM) is enabled on the VM	2020-06-01
<code>securityProfile.encryptionAtHost</code>	Identifies if <a href="#">Encryption at Host</a> is enabled on the VM	2021-11-01†
<code>securityProfile.securityType</code>	Identifies if the VM is a <a href="#">Trusted VM</a> or a <a href="#">Confidential VM</a>	2021-12-13†
<code>storageProfile</code>	See Storage Profile below	2019-06-01
<code>subscriptionId</code>	Azure subscription for the Virtual Machine	2017-08-01
<code>tags</code>	<a href="#">Tags</a> for your Virtual Machine	2017-08-01
<code>tagsList</code>	Tags formatted as a JSON array for easier programmatic parsing	2019-06-04
<code>userData</code>	The set of data specified when the VM was created for use during or after provisioning (Base64 encoded)	2021-01-01
<code>version</code>	Version of the VM image	2017-04-02
<code>virtualMachineScaleSet.id</code>	ID of the <a href="#">Virtual Machine Scale Set created with flexible orchestration</a> the Virtual Machine is part of. This field is not available for Virtual Machine Scale Sets created with uniform orchestration.	2021-03-01
<code>vmId</code>	<a href="#">Unique identifier</a> for the VM. The blog referenced only suits for VMs that have SMBIOS < 2.6. For VMs that have SMBIOS >= 2.6, the UUID from DMI is displayed in little-endian format, thus, there is no requirement to switch bytes.	2017-04-02
<code>vmScaleSetName</code>	<a href="#">Virtual machine scale set Name</a> of your virtual machine scale set	2017-12-01
<code>vmSize</code>	<a href="#">VM size</a>	2017-04-02
<code>zone</code>	<a href="#">Availability Zone</a> of your virtual machine	2017-12-01

† This version is not fully available yet and may not be supported in all regions.

## Storage profile

The storage profile of a VM is divided into three categories: image reference, OS disk, and data disks, plus an additional object for the local temporary disk.

The image reference object contains the following information about the OS image:

DATA	DESCRIPTION
<code>id</code>	Resource ID
<code>offer</code>	Offer of the platform or marketplace image
<code>publisher</code>	Image publisher
<code>sku</code>	Image sku
<code>version</code>	Version of the platform or marketplace image

The OS disk object contains the following information about the OS disk used by the VM:

DATA	DESCRIPTION
<code>caching</code>	Caching requirements
<code>createOption</code>	Information about how the VM was created
<code>diffDiskSettings</code>	Ephemeral disk settings
<code>diskSizeGB</code>	Size of the disk in GB
<code>image</code>	Source user image virtual hard disk
<code>managedDisk</code>	Managed disk parameters
<code>name</code>	Disk name
<code>vhd</code>	Virtual hard disk
<code>writeAcceleratorEnabled</code>	Whether or not writeAccelerator is enabled on the disk

The data disks array contains a list of data disks attached to the VM. Each data disk object contains the following information:

DATA	DESCRIPTION	VERSION INTRODUCED
<code>bytesPerSecondThrottle</code> *	Disk read/write quota in bytes	2021-05-01
<code>caching</code>	Caching requirements	2019-06-01
<code>createOption</code>	Information about how the VM was created	2019-06-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>diffDiskSettings</code>	Ephemeral disk settings	2019-06-01
<code>diskCapacityBytes</code> *	Size of disk in bytes	2021-05-01
<code>diskSizeGB</code>	Size of the disk in GB	2019-06-01
<code>encryptionSettings</code>	Encryption settings for the disk	2019-06-01
<code>image</code>	Source user image virtual hard disk	2019-06-01
<code>isSharedDisk</code> *	Identifies if the disk is shared between resources	2021-05-01
<code>isUltraDisk</code>	Identifies if the data disk is an Ultra Disk	2021-05-01
<code>lun</code>	Logical unit number of the disk	2019-06-01
<code>managedDisk</code>	Managed disk parameters	2019-06-01
<code>name</code>	Disk name	2019-06-01
<code>opsPerSecondThrottle</code> *	Disk read/write quota in IOPS	2021-05-01
<code>osType</code>	Type of OS included in the disk	2019-06-01
<code>vhd</code>	Virtual hard disk	2019-06-01
<code>writeAcceleratorEnabled</code>	Whether or not writeAccelerator is enabled on the disk	2019-06-01

\* These fields are only populated for Ultra Disks; they will be empty strings from non-Ultra Disks.

The encryption settings blob contains data about how the disk is encrypted (if it is encrypted):

DATA	DESCRIPTION	VERSION INTRODUCED
<code>diskEncryptionKey.sourceVault.id</code>	The location of the disk encryption key	2021-11-01†
<code>diskEncryptionKey.secretUrl</code>	The location of the secret	2021-11-01†
<code>keyEncryptionKey.sourceVault.id</code>	The location of the key encryption key	2021-11-01†
<code>keyEncryptionKey.keyUrl</code>	The location of the key	2021-11-01†

\† This version is not fully available yet and may not be supported in all regions.

The resource disk object contains the size of the [Local Temp Disk](#) attached to the VM, if it has one, in kilobytes. If there is [no local temp disk for the VM](#), this value is 0.

DATA	DESCRIPTION	VERSION INTRODUCED
<code>resourceDisk.size</code>	Size of the local temp disk for the VM (in kB)	2021-02-01

## Network

DATA	DESCRIPTION	VERSION INTRODUCED
<code>ipv4.privateIpAddress</code>	Local IPv4 address of the VM	2017-04-02
<code>ipv4.publicIpAddress</code>	Public IPv4 address of the VM	2017-04-02
<code>subnet.address</code>	Subnet address of the VM	2017-04-02
<code>subnet.prefix</code>	Subnet prefix, example 24	2017-04-02
<code>ipv6.ipAddress</code>	Local IPv6 address of the VM	2017-04-02
<code>macAddress</code>	VM mac address	2017-04-02

### NOTE

The nics returned by the network call are not guaranteed to be in order.

## Get user data

When creating a new VM, you can specify a set of data to be used during or after the VM provision, and retrieve it through IMDS. Check the end to end user data experience [here](#).

To set up user data, utilize the quickstart template [here](#). The sample below shows how to retrieve this data through IMDS. This feature is released with version `2021-01-01` and above.

### NOTE

Security notice: IMDS is open to all applications on the VM, sensitive data should not be placed in the user data.

- [Windows](#)
- [Linux](#)

```
$userData = Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri "http://169.254.169.254/metadata/instance/compute/userData?api-version=2021-01-01&format=text" [System.Text.Encoding]::UTF8.GetString([Convert]::FromBase64String($userData))
```

### Sample 1: Tracking VM running on Azure

As a service provider, you may require to track the number of VMs running your software or have agents that need to track uniqueness of the VM. To be able to get a unique ID for a VM, use the `vmId` field from Instance Metadata Service.

## Request

- [Windows](#)

- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/vmId?api-version=2017-08-01&format=text"
```

## Response

```
5c08b38e-4d57-4c23-ac45-aca61037f084
```

### Sample 2: Placement of different data replicas

For certain scenarios, placement of different data replicas is of prime importance. For example, [HDFS replica placement](#) or container placement via an [orchestrator](#) might require you to know the `platformFaultDomain` and `platformUpdateDomain` the VM is running on. You can also use [Availability Zones](#) for the instances to make these decisions. You can query this data directly via IMDS.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/platformFaultDomain?api-version=2017-08-01&format=text"
```

## Response

```
0
```

### Sample 3: Get VM tags

VM tags are included the instance API under `instance/compute/tags` endpoint. Tags may have been applied to your Azure VM to logically organize them into a taxonomy. The tags assigned to a VM can be retrieved by using the request below.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/tags?api-version=2017-08-01&format=text"
```

## Response

```
Department:IT;ReferenceNumber:123456;TestStatus:Pending
```

The `tags` field is a string with the tags delimited by semicolons. This output can be a problem if semicolons are used in the tags themselves. If a parser is written to programmatically extract the tags, you should rely on the `tagsList` field. The `tagsList` field is a JSON array with no delimiters, and consequently, easier to parse. The `tagsList` assigned to a VM can be retrieved by using the request below.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
"http://169.254.169.254/metadata/instance/compute/tagsList?api-version=2019-06-04" | ConvertTo-Json -Depth
64
```

## Response

- [Windows](#)
- [Linux](#)

```
{
  "value": [
    {
      "name": "Department",
      "value": "IT"
    },
    {
      "name": "ReferenceNumber",
      "value": "123456"
    },
    {
      "name": "TestStatus",
      "value": "Pending"
    }
  ],
  "Count": 3
}
```

## Sample 4: Get more information about the VM during support case

As a service provider, you may get a support call where you would like to know more information about the VM. Asking the customer to share the compute metadata can provide basic information for the support professional to know about the kind of VM on Azure.

## Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
"http://169.254.169.254/metadata/instance/compute?api-version=2020-09-01" | ConvertTo-Json -Depth 64
```

## Response

### NOTE

The response is a JSON string. The following example response is pretty-printed for readability.

- [Windows](#)
- [Linux](#)

```
{
  "azEnvironment": "AZUREPUBLICCLOUD",
```

```
"extendedLocation": {
    "type": "edgeZone",
    "name": "microsoftlosangeles"
},
"evictionPolicy": "",
"additionalCapabilities": {
    "hibernationEnabled": "false"
},
"hostGroup": {
    "id": "testHostGroupId"
},
"isHostCompatibilityLayerVm": "true",
"licenseType": "Windows_Client",
"location": "westus",
"name": "examplevmname",
"offer": "WindowsServer",
"osProfile": {
    "adminUsername": "admin",
    "computerName": "examplevmname",
    "disablePasswordAuthentication": "true"
},
"osType": "Windows",
"placementGroupId": "f67c14ab-e92c-408c-ae2d-da15866ec79a",
"plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
},
"platformFaultDomain": "36",
"platformUpdateDomain": "42",
"priority": "Regular",
"publicKeys": [
    {
        "keyData": "ssh-rsa 0",
        "path": "/home/user/.ssh/authorized_keys0"
    },
    {
        "keyData": "ssh-rsa 1",
        "path": "/home/user/.ssh/authorized_keys1"
    }
],
"publisher": "RDDE-Test-Microsoft-Windows-Server-Group",
"resourceGroupName": "macikgo-test-may-23",
"resourceId": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/Microsoft.Compute/virtualMachines/examplevmname",
"securityProfile": {
    "secureBootEnabled": "true",
    "virtualTpmEnabled": "false",
    "encryptionAtHost": "true",
    "securityType": "TrustedLaunch"
},
"sku": "2019-Datacenter",
"storageProfile": {
    "dataDisks": [
        {
            "bytesPerSecondThrottle": "979202048",
            "caching": "None",
            "createOption": "Empty",
            "diskCapacityBytes": "274877906944",
            "diskSizeGB": "1024",
            "image": {
                "uri": ""
            },
            "isSharedDisk": "false",
            "isUltraDisk": "true",
            "lun": "0",
            "managedDisk": {
                "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-23/providers/MicrosoftCompute/disks/exampledatadiskname",
                "storageAccountType": "StandardSSD_LRS"
            }
        }
    ]
}
```

```
        "name": "exampleddiskname",
        "opsPerSecondThrottle": "65280",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    }],
    "imageReference": {
        "id": "",
        "offer": "WindowsServer",
        "publisher": "MicrosoftWindowsServer",
        "sku": "2019-Datacenter",
        "version": "latest"
    },
    "osDisk": {
        "caching": "ReadWrite",
        "createOption": "FromImage",
        "diskSizeGB": "30",
        "diffDiskSettings": {
            "option": "Local"
        },
        "encryptionSettings": {
            "enabled": "false",
            "diskEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-source-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "secretUrl": "https://test-disk.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-
xxxxxxxxxx/xxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxx"
            },
            "keyEncryptionKey": {
                "sourceVault": {
                    "id": "/subscriptions/test-key-
guid/resourceGroups/testrg/providers/Microsoft.KeyVault/vaults/test-kv"
                },
                "keyUrl": "https://test-key.vault.azure.net/secrets/xxxxx-xxxx-xxxx-xxxx-xxxx-
xxxx-xxxx-xxxx-xxxxxxxxxx"
            }
        },
        "image": {
            "uri": ""
        },
        "managedDisk": {
            "id": "/subscriptions/xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx/resourceGroups/macikgo-test-may-
23/providers/Microsoft.Compute/disks/exampleosdiskname",
            "storageAccountType": "StandardSSD_LRS"
        },
        "name": "exampleosdiskname",
        "osType": "Windows",
        "vhd": {
            "uri": ""
        },
        "writeAcceleratorEnabled": "false"
    },
    "resourceDisk": {
        "size": "4096"
    }
},
"subscriptionId": "xxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx",
"tags": "baz:bash;foo:bar",
"version": "15.05.22",
"virtualMachineScaleSet": {
    "id": "/subscriptions/xxxxxxxx-xxxx-xxx-xxx-xxxx/resourceGroups/resource-group-
name/providers/Microsoft.Compute/virtualMachineScaleSets/virtual-machine-scale-set-name"
},
"vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6",
"vmScaleSetName": "crpteste9vflji9",
"vmSize": "Standard A3".
```

```
        "zone": ""  
    }  
}
```

#### Sample 5: Get the Azure Environment where the VM is running

Azure has various sovereign clouds like [Azure Government](#). Sometimes you need the Azure Environment to make some runtime decisions. The following sample shows you how you can achieve this behavior.

##### Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/compute/azEnvironment?api-version=2018-10-01&format=text"
```

##### Response

```
AzurePublicCloud
```

The cloud and the values of the Azure environment are listed here.

CLOUD	AZURE ENVIRONMENT
All generally available global Azure regions	AzurePublicCloud
Azure Government	AzureUSGovernmentCloud
Azure China 21Vianet	AzureChinaCloud
Azure Germany	AzureGermanCloud

#### Sample 6: Retrieve network information

##### Request

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri  
"http://169.254.169.254/metadata/instance/network?api-version=2017-08-01" | ConvertTo-Json -Depth 64
```

##### Response

```
{
  "interface": [
    {
      "ipv4": {
        "ipAddress": [
          {
            "privateIpAddress": "10.1.0.4",
            "publicIpAddress": "X.X.X.X"
          }
        ],
        "subnet": [
          {
            "address": "10.1.0.0",
            "prefix": "24"
          }
        ]
      },
      "ipv6": {
        "ipAddress": []
      },
      "macAddress": "000D3AF806EC"
    }
  ]
}
```

#### Sample 7: Retrieve public IP address

- [Windows](#)
- [Linux](#)

```
Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
"http://169.254.169.254/metadata/instance/network/interface/0/ipv4/ipAddress/0/publicIpAddress?api-
version=2017-08-01&format=text"
```

#### NOTE

- If you are looking to retrieve IMDS information for **Standard** SKU Public IP address, review [Load Balancer Metadata API](#) for more infomration.

## Attested data

### Get Attested data

IMDS helps to provide guarantees that the data provided is coming from Azure. Microsoft signs part of this information, so you can confirm that an image in Azure Marketplace is the one you are running on Azure.

```
GET /metadata/attested/document
```

#### Parameters

NAME	REQUIRED/OPTIONAL	DESCRIPTION
<code>api-version</code>	Required	The version used to service the request.

Name	Required/Optional	Description
nonce	Optional	A 10-digit string that serves as a cryptographic nonce. If no value is provided, IMDS uses the current UTC timestamp.

## Response

```
{
  "encoding": "pkcs7",

  "signature": "MIIEEgYJKoZIhvcNAQcCoIIEAzCCA/8CAQExDzANBgkqhkiG9w0BAQsFADCBugYJKoZIhvcNAQcBoIGsBIGpeyJub25jZSI
6IjEyMzQ1NjY3NjYiLCJwbGFuIjp7Im5hbWUiOiIiLCJwcm9kdWN0IjoiiIwicHVibGlzaGVyIjoiIn0sInRpblVtDGFtcCI6eyJjcmVhdGV
kt24i0IiXMS8yMC8xOCAYmjowNzozOSAtMDAwMCIsImV4cGlyZXNPbiI6IjExLzIwLzE4IDIyOjA40jI0IC0wMDAwIn0sInZtSWQiOiiifaC
CAj8wggI7MIIbPkadAgEcAhBnxW5Kh8ds1EBA0E2mIBj0MA0GCSqGSIB3DQEBAUAMCsxKTAAnBGNVBAMTIHrlc3RzdWJkb21haW4ubWv0YWR
hdGEuYXp1cmUuY29tMB4XDTE4MTEyMDIxNTc1N1oXDTE4MTIyMDIxNTc1NlowKzEpMcC GA1UEAxMgdGVzdHN1YmRvbWFpbis5tZRhZGF0YS5
henVz5jb20wgZ8wDQYJKoZIhvcNAQEBQAdgY0AMIGJAoGBAML/tBo86ENWPzmXZ0kPkX5dY5QZ150ma8lommse71x2sCLonzv4/Ulk4H
+jMMWRRwIea2CuQ5RhdWAhvKq6if4okNt66fxm+YTVz9z0CTfCmLT+nsdf0AsG1xZppEapC0Cd9vD6NCKyE8aYI1pliaeOnFjG0WvMY04u
Wz2MdAgMBAAGjYDBeMFwGA1UdAQRVMFOAEnYkHLa04Ut4Mpt7TkJFFyhLTArMSkwJwYDVQQDEyB0ZXN0c3ViZG9tYwluLm1ldGFkYXRhLmF
6dXJ1LmNvbYIQZ8VuSofHbJRAQNBnpiAsdDANBgkqhkiG9w0BAQFAAOBgQCLSM6aX5Bs1KHCrp4VQtzPzXF71rVKCocHy3N9PTJQ9Fpnd+
bYw2vSpQHg/AiG82WuDFpPreJvr7Pa938mZqW9HUOGjQKK2FYDTg6FXD8pkPdygh1X5boGWAMMrf7bFkup+1sT+n2tRw2wbNkn01tQ0wICq
y2VqzWwLi45RBwTGB6DCB5QIBATA/MCsxtAnBGNVBAMTIHrlc3RzdWJkb21haW4ubWv0YWRhdGEuYXp1cmUuY29tAhBnxW5Kh8ds1EBA0E2
mIBj0MA0GCSqGSIB3DQEBCwUAMA0GCSqGSIB3DQEBAQUABIGAl1d1BM/yYIqqv8SDE4kjQo3U1/IKAVR8ETKcve5BAdGSNkTUooUGVniTxev
Dj5NkmazOaKZp9fEtByqqPOyw/n1xaZg0044HDG1PUJ90xVYmfek6p9RpJBu6kiKhnnYTelUk5u75phe5ZbMFbhupPhXmYAdjc7Nmw97nx8N
nprQ="
}
```

The signature blob is a [pkcs7](#)-signed version of document. It contains the certificate used for signing along with certain VM-specific details.

For VMs created by using Azure Resource Manager, the document includes `vmId`, `sku`, `nonce`, `subscriptionId`, `timeStamp` for creation and expiry of the document, and the plan information about the image. The plan information is only populated for Azure Marketplace images.

For VMs created by using the classic deployment model, only the `vmId` and `subscriptionId` are guaranteed to be populated. You can extract the certificate from the response, and use it to confirm that the response is valid and is coming from Azure.

The decoded document contains the following fields:

Data	Description	Version Introduced
<code>licenseType</code>	Type of license for <a href="#">Azure Hybrid Benefit</a> . This is only present for AHB-enabled VMs.	2020-09-01
<code>nonce</code>	A string that can be optionally provided with the request. If no <code>nonce</code> was supplied, the current Coordinated Universal Time timestamp is used.	2018-10-01
<code>plan</code>	The <a href="#">Azure Marketplace Image plan</a> . Contains the plan ID (name), product image or offer (product), and publisher ID (publisher).	2018-10-01

DATA	DESCRIPTION	VERSION INTRODUCED
<code>timestamp.createdOn</code>	The UTC timestamp for when the signed document was created	2018-20-01
<code>timestamp.expiresOn</code>	The UTC timestamp for when the signed document expires	2018-10-01
<code>vmId</code>	Unique identifier for the VM	2018-10-01
<code>subscriptionId</code>	Azure subscription for the Virtual Machine	2019-04-30
<code>sku</code>	Specific SKU for the VM image (correlates to <code>compute/sku</code> property from the Instance Metadata endpoint [ <code>/metadata/instance</code> ])	2019-11-01

#### NOTE

For Classic (non-Azure Resource Manager) VMs, only the `vmId` is guaranteed to be populated.

Example document:

```
{
  "nonce": "20201130-211924",
  "plan": {
    "name": "planName",
    "product": "planProduct",
    "publisher": "planPublisher"
  },
  "sku": "Windows-Server-2012-R2-Datacenter",
  "subscriptionId": "8d10da13-8125-4ba9-a717-bf7490507b3d",
  "timeStamp": {
    "createdOn": "11/30/20 21:19:19 -0000",
    "expiresOn": "11/30/20 21:19:24 -0000"
  },
  "vmId": "02aab8a4-74ef-476e-8182-f6d2ba4166a6"
}
```

#### Sample 1: Validate that the VM is running in Azure

Vendors in Azure Marketplace want to ensure that their software is licensed to run only in Azure. If someone copies the VHD to an on-premises environment, the vendor needs to be able to detect that. Through IMDS, these vendors can get signed data that guarantees response only from Azure.

#### NOTE

This sample requires the jq utility to be installed.

## Validation

- [Windows](#)
- [Linux](#)

```

# Get the signature
$attestedDoc = Invoke-RestMethod -Headers @{"Metadata"="true"} -Method GET -NoProxy -Uri
http://169.254.169.254/metadata/attested/document?api-version=2020-09-01
# Decode the signature
$signature = [System.Convert]::FromBase64String($attestedDoc.signature)

```

Verify that the signature is from Microsoft Azure and check the certificate chain for errors.

```

# Get certificate chain
$cert = [System.Security.Cryptography.X509Certificates.X509Certificate2]($signature)
$chain = New-Object -TypeName System.Security.Cryptography.X509Certificates.X509Chain
$chain.Build($cert)
# Print the Subject of each certificate in the chain
foreach($element in $chain.ChainElements)
{
    Write-Host $element.Certificate.Subject
}

# Get the content of the signed document
Add-Type -AssemblyName System.Security
$signedCms = New-Object -TypeName System.Security.Cryptography.Pkcs.SignedCms
$signedCms.Decode($signature);
$content = [System.Text.Encoding]::UTF8.GetString($signedCms.ContentInfo.Content)
Write-Host "Attested data: " $content
$json = $content | ConvertFrom-Json
# Do additional validation here

```

#### NOTE

Due to IMDS's caching mechanism, a previously cached `nonce` value might be returned.

The `nonce` in the signed document can be compared if you provided a `nonce` parameter in the initial request.

#### NOTE

The certificate for the public cloud and each sovereign cloud will be different.

CLOUD	CERTIFICATE
All generally available global Azure regions	*.metadata.azure.com
Azure Government	*.metadata.azure.us
Azure China 21Vianet	*.metadata.azure.cn
Azure Germany	*.metadata.microsoftazure.de

#### NOTE

The certificates might not have an exact match of `metadata.azure.com` for the public cloud. For this reason, the certification validation should allow a common name from any `.metadata.azure.com` subdomain.

In cases where the intermediate certificate can't be downloaded due to network constraints during validation, you can pin the intermediate certificate. Azure rolls over the certificates, which is standard PKI practice. You must

update the pinned certificates when rollover happens. Whenever a change to update the intermediate certificate is planned, the Azure blog is updated, and Azure customers are notified.

You can find the intermediate certificates on [this page](#). The intermediate certificates for each of the regions can be different.

#### NOTE

The intermediate certificate for Azure China 21Vianet will be from DigiCert Global Root CA, instead of Baltimore. If you pinned the intermediate certificates for Azure China as part of a root chain authority change, the intermediate certificates must be updated.

#### NOTE

Starting February 2022, our Attested Data certificates will be impacted by a TLS change. Due to this, the root CA will change from Baltimore CyberTrust to DigiCert Global G2 only for Public and US Government clouds. If you have the Baltimore CyberTrust cert or other intermediate certificates listed in [this post](#) pinned, please follow the instructions listed there **immediately** to prevent any disruptions from using the Attested Data endpoint.

## Managed identity

A managed identity, assigned by the system, can be enabled on the VM. You can also assign one or more user-assigned managed identities to the VM. You can then request tokens for managed identities from IMDS. Use these tokens to authenticate with other Azure services, such as Azure Key Vault.

For detailed steps to enable this feature, see [Acquire an access token](#).

## Load Balancer Metadata

When you place virtual machine or virtual machine set instances behind an Azure Standard Load Balancer, you can use IMDS to retrieve metadata related to the load balancer and the instances. For more information, see [Retrieve load balancer information](#).

## Scheduled events

You can obtain the status of the scheduled events by using IMDS. Then the user can specify a set of actions to run upon these events. For more information, see [Scheduled events for Linux](#) or [Scheduled events for Windows](#).

## Sample code in different languages

The following table lists samples of calling IMDS by using different languages inside the VM:

LANGUAGE	EXAMPLE
Bash	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.sh">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.sh</a>
C#	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.cs">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.cs</a>
Go	<a href="https://github.com/Microsoft/azureimds/blob/master/imdssample.go">https://github.com/Microsoft/azureimds/blob/master/imdssample.go</a>

LANGUAGE	EXAMPLE
Java	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.java">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.java</a>
NodeJS	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.js">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.js</a>
Perl	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.pl">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.pl</a>
PowerShell	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.ps1">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.ps1</a>
Puppet	<a href="https://github.com/keirans/azurometadata">https://github.com/keirans/azurometadata</a>
Python	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.py">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.py</a>
Ruby	<a href="https://github.com/Microsoft/azureimds/blob/master/IMDSSample.rb">https://github.com/Microsoft/azureimds/blob/master/IMDSSample.rb</a>

## Errors and debugging

If there is a data element not found or a malformed request, the Instance Metadata Service returns standard HTTP errors. For example:

HTTP STATUS CODE	REASON
200 OK	The request was successful.
400 Bad Request	Missing <code>Metadata: true</code> header or missing parameter <code>format=json</code> when querying a leaf node
404 Not Found	The requested element doesn't exist
405 Method Not Allowed	The HTTP method (verb) is not supported on the endpoint.
410 Gone	Retry after some time for a max of 70 seconds
429 Too Many Requests	API <a href="#">Rate Limits</a> have been exceeded
500 Service Error	Retry after some time

## Frequently asked questions

- I am getting the error `400 Bad Request, Required metadata header not specified`. What does this mean?
  - IMDS requires the header `Metadata: true` to be passed in the request. Passing this header in the REST call allows access to IMDS.
- Why am I not getting compute information for my VM?
  - Currently, IMDS only supports instances created with Azure Resource Manager.

- I created my VM through Azure Resource Manager some time ago. Why am I not seeing compute metadata information?
  - If you created your VM after September 2016, add a [tag](#) to start seeing compute metadata. If you created your VM before September 2016, add or remove extensions or data disks to the VM instance to refresh metadata.
- Is user data the same as custom data?
  - User data offers the similar functionality to custom data, allowing you to pass your own metadata to the VM instance. The difference is, user data is retrieved through IMDS, and is persistent throughout the lifetime of the VM instance. Existing custom data feature will continue to work as described in [this article](#). However you can only get custom data through local system folder, not through IMDS.
- Why am I not seeing all data populated for a new version?
  - If you created your VM after September 2016, add a [tag](#) to start seeing compute metadata. If you created your VM before September 2016, add or remove extensions or data disks to the VM instance to refresh metadata.
- Why am I getting the error `500 Internal Server Error` or `410 Resource Gone ?`?
  - Retry your request. For more information, see [Transient fault handling](#). If the problem persists, create a support issue in the Azure portal for the VM.
- Would this work for virtual machine scale set instances?
  - Yes, IMDS is available for virtual machine scale set instances.
- I updated my tags in virtual machine scale sets, but they don't appear in the instances (unlike single instance VMs). Am I doing something wrong?
  - Currently tags for virtual machine scale sets only show to the VM on a reboot, reimagine, or disk change to the instance.
- Why am I am not seeing the SKU information for my VM in `instance/compute` details?
  - For custom images created from Azure Marketplace, Azure platform doesn't retain the SKU information for the custom image and the details for any VMs created from the custom image. This is by design and hence not surfaced in the VM `instance/compute` details.
- Why is my request timed out for my call to the service?
  - Metadata calls must be made from the primary IP address assigned to the primary network card of the VM. Additionally, if you've changed your routes, there must be a route for the `169.254.169.254/32` address in your VM's local routing table.
    - [Windows](#)
    - [Linux](#)

1. Dump your local routing table and look for the IMDS entry. For example:

```

> route print
IPv4 Route Table
=====
Active Routes:
Network Destination      Netmask        Gateway       Interface Metric
          0.0.0.0        0.0.0.0    172.16.69.1   172.16.69.7    10
         127.0.0.0    255.0.0.0     On-link        127.0.0.1    331
         127.0.0.1  255.255.255.255     On-link        127.0.0.1    331
  127.255.255.255  255.255.255.255     On-link        127.0.0.1    331
  168.63.129.16  255.255.255.255    172.16.69.1   172.16.69.7    11
 169.254.169.254  255.255.255.255    172.16.69.1   172.16.69.7    11
... (continues) ...

```

2. Verify that a route exists for `169.254.169.254`, and note the corresponding network interface (for example, `172.16.69.7`).
3. Dump the interface configuration and find the interface that corresponds to the one referenced in the routing table, noting the MAC (physical) address.

```

> ipconfig /all
... (continues) ...
Ethernet adapter Ethernet:

Connection-specific DNS Suffix . : xic3mnxjiefupcwr1mcs1rjqiqa.cx.internal.cloudapp.net
Description . . . . . : Microsoft Hyper-V Network Adapter
Physical Address. . . . . : 00-0D-3A-E5-1C-C0
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . : fe80::3166:ce5a:2bd5:a6d1%3(Preferred)
IPv4 Address. . . . . : 172.16.69.7(Preferred)
Subnet Mask . . . . . : 255.255.255.0
... (continues) ...

```

4. Confirm that the interface corresponds to the VM's primary NIC and primary IP. You can find the primary NIC and IP by looking at the network configuration in the Azure portal, or by looking it up with the Azure CLI. Note the private IPs (and the MAC address if you're using the CLI). Here's a PowerShell CLI example:

```

$ResourceGroup = '<Resource_Group>'
$VmName = '<VM_Name>'
$NicNames = az vm nic list --resource-group $ResourceGroup --vm-name $VmName |
ConvertFrom-Json | Foreach-Object { $_.id.Split('/')[-1] }
foreach($NicName in $NicNames)
{
    $Nic = az vm nic show --resource-group $ResourceGroup --vm-name $VmName --nic
    $NicName | ConvertFrom-Json
    Write-Host $NicName, $Nic.primary, $Nic.macAddress
}
# Output: wintest767 True 00-0D-3A-E5-1C-C0

```

5. If they don't match, update the routing table so that the primary NIC and IP are targeted.

- Failover clustering in Windows Server
  - When you're querying IMDS with failover clustering, it's sometimes necessary to add a route to the routing table. Here's how:
    1. Open a command prompt with administrator privileges.
    2. Run the following command, and note the address of the Interface for Network Destination (`0.0.0.0`) in the IPv4 Route Table.

```
route print
```

#### NOTE

The following example output is from a Windows Server VM with failover cluster enabled. For simplicity, the output contains only the IPv4 Route Table.

#### IPv4 Route Table

Active Routes:					
Network Destination	Netmask	Gateway	Interface	Metric	
0.0.0.0	0.0.0.0	10.0.1.1	10.0.1.10	266	
10.0.1.0	255.255.255.192	On-link	10.0.1.10	266	
10.0.1.10	255.255.255.255	On-link	10.0.1.10	266	
10.0.1.15	255.255.255.255	On-link	10.0.1.10	266	
10.0.1.63	255.255.255.255	On-link	10.0.1.10	266	
127.0.0.0	255.0.0.0	On-link	127.0.0.1	331	
127.0.0.1	255.255.255.255	On-link	127.0.0.1	331	
127.255.255.255	255.255.255.255	On-link	127.0.0.1	331	
169.254.0.0	255.255.0.0	On-link	169.254.1.156	271	
169.254.1.156	255.255.255.255	On-link	169.254.1.156	271	
169.254.255.255	255.255.255.255	On-link	169.254.1.156	271	
224.0.0.0	240.0.0.0	On-link	127.0.0.1	331	
224.0.0.0	240.0.0.0	On-link	169.254.1.156	271	
255.255.255.255	255.255.255.255	On-link	127.0.0.1	331	
255.255.255.255	255.255.255.255	On-link	169.254.1.156	271	
255.255.255.255	255.255.255.255	On-link	10.0.1.10	266	

Run the following command and use the address of the Interface for Network Destination ( `0.0.0.0` ), which is ( `10.0.1.10` ) in this example.

```
route add 169.254.169.254/32 10.0.1.10 metric 1 -p
```

## Support

If you aren't able to get a metadata response after multiple attempts, you can create a support issue in the Azure portal.

## Product feedback

You can provide product feedback and ideas to our user feedback channel under Virtual Machines > Instance Metadata Service [here](#)

## Next steps

- [Acquire an access token for the VM](#)
- [Scheduled events for Linux](#)
- [Scheduled events for Windows](#)

# Get Virtual Machine usage metrics using the REST API

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This example shows how to retrieve the CPU usage for a Linux Virtual Machine using the [Azure REST API](#).

Complete reference documentation and additional samples for the REST API are available in the [Azure Monitor REST reference](#).

## Build the request

Use the following GET request to collect the [Percentage CPU metric](#) from a Virtual Machine

```
GET  
https://management.azure.com/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmname}/providers/microsoft.insights/metrics?api-version=2018-01-01&metricnames=Percentage%20CPU&timespan=2018-06-05T03:00:00Z/2018-06-07T03:00:00Z
```

### Request headers

The following headers are required:

REQUEST HEADER	DESCRIPTION
<i>Content-Type</i> :	Required. Set to <code>application/json</code> .
<i>Authorization</i> :	Required. Set to a valid <code>Bearer</code> access token.

### URI parameters

NAME	DESCRIPTION
subscriptionId	The subscription ID that identifies an Azure subscription. If you have multiple subscriptions, see <a href="#">Working with multiple subscriptions</a> .
resourceGroupName	The name of the Azure resource group associated with the resource. You can get this value from the Azure Resource Manager API, CLI, or the portal.
vmname	The name of the Azure Virtual Machine.
metricnames	Comma-separated list of valid <a href="#">Load Balancer metrics</a> .
api-version	The API version to use for the request.  This document covers api-version <code>2018-01-01</code> , included in the above URL.

NAME	DESCRIPTION
timespan	String with the following format <code>startDateTime_ISO/endDateTime_ISO</code> that defines the time range of the returned metrics. This optional parameter is set to return a day's worth of data in the example.

## Request body

No request body is needed for this operation.

## Handle the response

Status code 200 is returned when the list of metric values is returned successfully. A full list of error codes is available in the [reference documentation](#).

## Example response

```
{
  "cost": 0,
  "timespan": "2018-06-08T23:48:10Z/2018-06-09T00:48:10Z",
  "interval": "PT1M",
  "value": [
    {
      "id": "/subscriptions/{subscriptionId}/resourceGroups/{resourceGroupName}/providers/Microsoft.Compute/virtualMachines/{vmname}/providers/microsoft.insights/metrics?api-version=2018-01-01&metricnames=Percentage%20CPU",
      "type": "Microsoft.Insights/metrics",
      "name": {
        "value": "Percentage CPU",
        "localizedValue": "Percentage CPU"
      },
      "unit": "Percent",
      "timeseries": [
        {
          "metadatavalues": [],
          "data": [
            {
              "timeStamp": "2018-06-08T23:48:00Z",
              "average": 0.44
            },
            {
              "timeStamp": "2018-06-08T23:49:00Z",
              "average": 0.31
            },
            {
              "timeStamp": "2018-06-08T23:50:00Z",
              "average": 0.29
            },
            {
              "timeStamp": "2018-06-08T23:51:00Z",
              "average": 0.29
            },
            {
              "timeStamp": "2018-06-08T23:52:00Z",
              "average": 0.285
            }
          ]
        }
      ]
    }
  ]
}
```

# Azure boot diagnostics

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Boot diagnostics is a debugging feature for Azure virtual machines (VM) that allows diagnosis of VM boot failures. Boot diagnostics enables a user to observe the state of their VM as it is booting up by collecting serial log information and screenshots.

## Boot diagnostics storage account

When you create a VM in Azure portal, boot diagnostics is enabled by default. The recommended boot diagnostics experience is to use a managed storage account, as it yields significant performance improvements in the time to create an Azure VM. This is because an Azure managed storage account will be used, removing the time it takes to create a new user storage account to store the boot diagnostics data.

### IMPORTANT

The boot diagnostics data blobs (which comprise of logs and snapshot images) are stored in a managed storage account. Customers will be charged only on used GiBs by the blobs, not on the disk's provisioned size. The snapshot meters will be used for billing of the managed storage account. Because the managed accounts are created on either Standard LRS or Standard ZRS, customers will be charged at \$0.05/GB per month for the size of their diagnostic data blobs only. For more information on this pricing, see [Managed disks pricing](#). Customers will see this charge tied to their VM resource URI.

An alternative boot diagnostic experience is to use a custom storage account. A user can either create a new storage account or use an existing one. When the storage firewall is enabled on the custom storage account (**Enabled from all networks** option isn't selected), you must:

- Make sure that access through the storage firewall is allowed for the Azure platform to publish the screenshot and serial log. To do this, go to the custom boot diagnostics storage account in the Azure portal and then select **Networking** from the **Security + networking** section. Check if the **Allow Azure services on the trusted services list to access this storage account** checkbox is selected.
- Allow storage firewall for users to view the boot screenshots or serial logs. To do this, add your network or the client/browser's Internet IPs as firewall exclusions. For more information, see [Configure Azure Storage firewalls and virtual networks](#).

To configure the storage firewall for Azure Serial Console, see [Use Serial Console with custom boot diagnostics storage account firewall enabled](#).

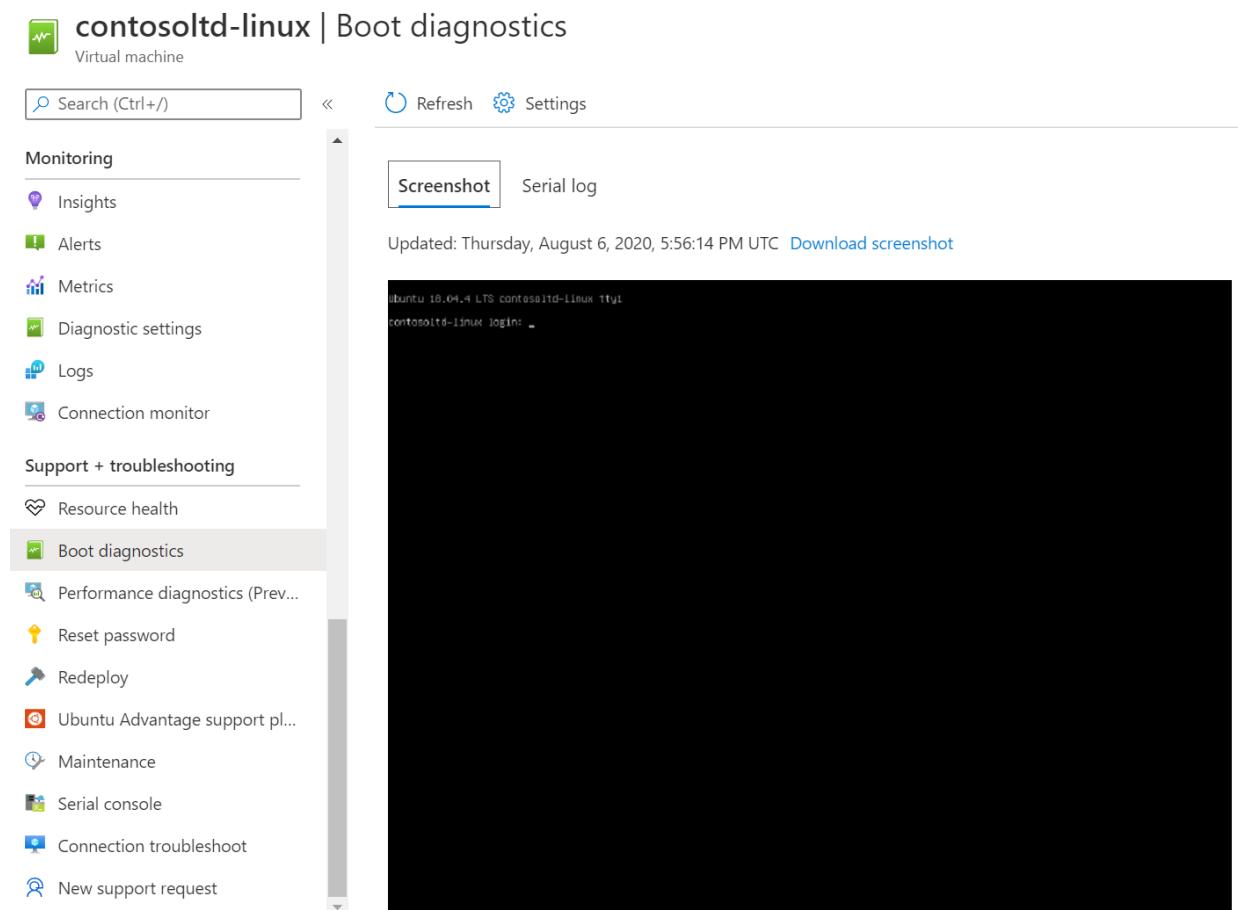
### NOTE

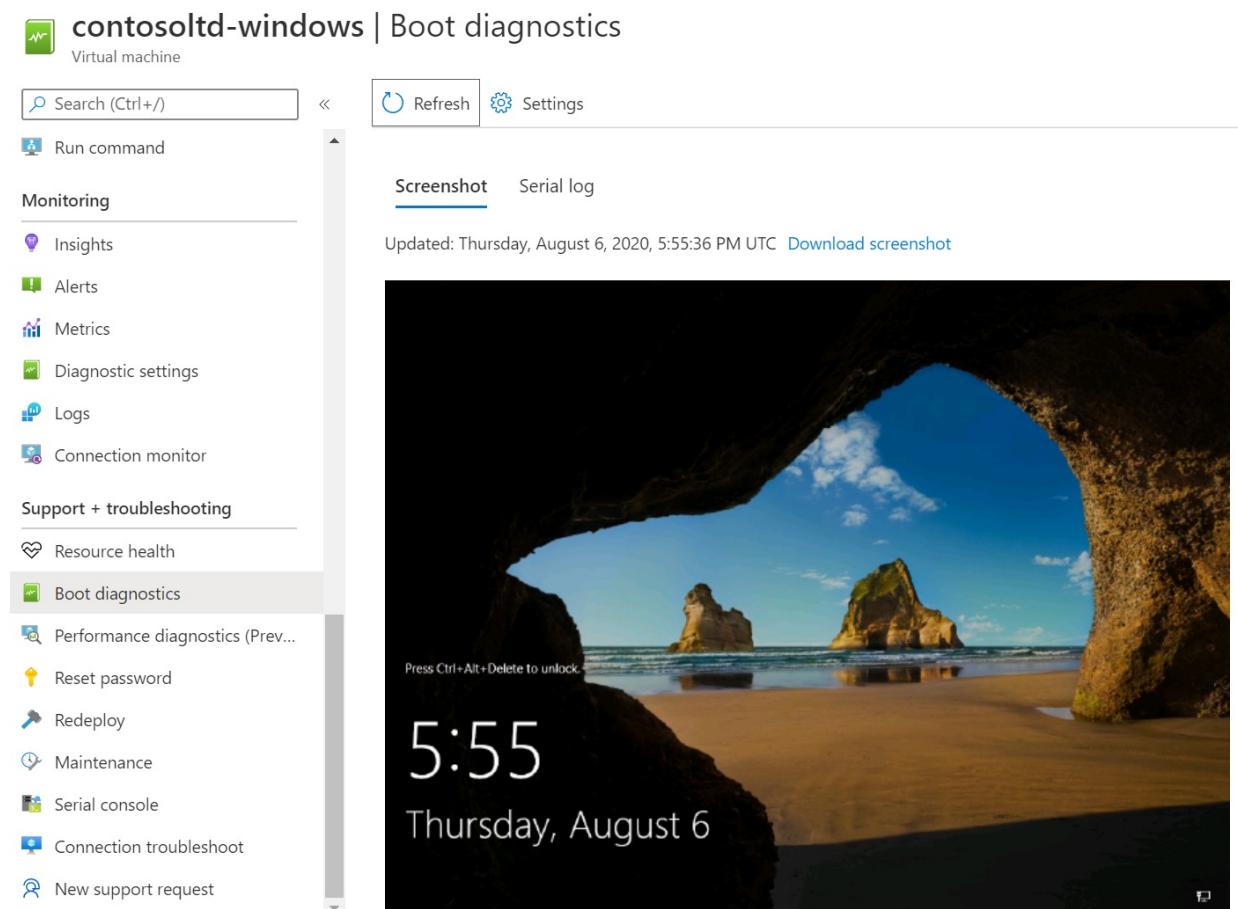
The custom storage account associated with boot diagnostics requires the storage account and the associated virtual machines reside in the same region and subscription.

## Boot diagnostics view

Go to the virtual machine blade in the Azure portal, the boot diagnostics option is under the *Support and Troubleshooting* section in the Azure portal. Selecting boot diagnostics will display a screenshot and serial log information. The serial log contains kernel messaging and the screenshot is a snapshot of your VMs current

state. Based on if the VM is running Windows or Linux determines what the expected screenshot would look like. For Windows, users will see a desktop background and for Linux, users will see a login prompt.





## Enable managed boot diagnostics

Managed boot diagnostics can be enabled through the Azure portal, CLI and ARM Templates.

### Enable managed boot diagnostics using the Azure portal

When you create a VM in the Azure portal, the default setting is to have boot diagnostics enabled using a managed storage account. To view this, navigate to the *Management* tab during the VM creation.

## Create a virtual machine

Basics Disks Networking **Management** Advanced Tags Review + create

Configure monitoring and management options for your VM.

**Azure Security Center**

Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads. [Learn more](#)

Your subscription is protected by Azure Security Center basic plan.

**Monitoring**

Enable detailed monitoring  On  Off

Boot diagnostics  Enable with managed storage account (recommended)  Enable with custom storage account  Disable

OS guest diagnostics  On  Off

### Enable managed boot diagnostics using CLI

Boot diagnostics with a managed storage account is supported in Azure CLI 2.12.0 and later. If you don't input a name or URI for a storage account, a managed account will be used. For more information and code samples, see the [CLI documentation for boot diagnostics](#).

### Enable managed boot diagnostics using PowerShell

Boot diagnostics with a managed storage account is supported in Azure PowerShell 6.6.0 and later. If you don't input a name or URI for a storage account, a managed account will be used. For more information and code samples, see the [PowerShell documentation for boot diagnostics](#).

### Enable managed boot diagnostics using Azure Resource Manager (ARM) templates

Everything after API version 2020-06-01 supports managed boot diagnostics. For more information, see [boot diagnostics instance view](#).

```

    "name": "[parameters('virtualMachineName')]",
    "type": "Microsoft.Compute/virtualMachines",
    "apiVersion": "2020-06-01",
    "location": "[parameters('location')]",
    "dependsOn": [
        "[concat('Microsoft.Network/networkInterfaces/', parameters('networkInterfaceName'))]"
    ],
    "properties": {
        "hardwareProfile": {
            "vmSize": "[parameters('virtualMachineSize')]"
        },
        "storageProfile": {
            "osDisk": {
                "createOption": "fromImage",
                "managedDisk": {
                    "storageAccountType": "[parameters('osDiskType')]"
                }
            },
            "imageReference": {
                "publisher": "Canonical",
                "offer": "UbuntuServer",
                "sku": "18.04-LTS",
                "version": "latest"
            }
        },
        "networkProfile": {
            "networkInterfaces": [
                {
                    "id": "[resourceId('Microsoft.Network/networkInterfaces',
parameters('networkInterfaceName'))]"
                }
            ]
        },
        "osProfile": {
            "computerName": "[parameters('virtualMachineComputerName')]",
            "adminUsername": "[parameters('adminUsername')]",
            "linuxConfiguration": {
                "disablePasswordAuthentication": true
            },
            "diagnosticsProfile": {
                "bootDiagnostics": {
                    "enabled": true
                }
            }
        }
    }
],

```

## Limitations

- Managed boot diagnostics is only available for Azure Resource Manager VMs.
- Managed boot diagnostics doesn't support VMs using unmanaged OS disks.
- Boot diagnostics doesn't support premium storage accounts or zone redundant storage accounts. If either of these are used for boot diagnostics users will receive an `StorageAccountTypeNotSupported` error when starting the VM.
- Managed storage accounts are supported in Resource Manager API version "2020-06-01" and later.
- Portal only supports the use of boot diagnostics with a managed storage account for single instance VMs.
- Users cannot configure a retention period for Managed Boot Diagnostics. The logs will be overwritten when the total size crosses 1 GB.

## Next steps

Learn more about the [Azure Serial Console](#) and how to use boot diagnostics to [troubleshoot virtual machines in Azure](#).

# Backup and restore options for virtual machines in Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

You can protect your data by taking backups at regular intervals. There are several backup options available for virtual machines (VMs), depending on your use-case.

## Azure Backup

You'll use Azure Backup for most use-cases involving backup operations on Azure VMs running production workloads. Azure Backup supports application-consistent backups for both Windows and Linux VMs. Azure Backup creates recovery points that are stored in geo-redundant recovery vaults. When you restore from a recovery point, you can restore entire VM or specific files.

For a simple, hands-on introduction to Azure Backup for Azure VMs, see the [Azure Backup quickstart](#).

For more information on how Azure Backup works, see [Plan your VM backup infrastructure in Azure](#)

## Azure Site Recovery

Azure Site Recovery protects your VMs from a major disaster scenario. These scenarios may include widespread service interruptions or regional outages caused by natural disasters. You can configure Azure Site Recovery for your VMs so that your applications are recoverable in matter of minutes with a single click. You can replicate to an Azure region of your choice, since recovery isn't restricted to paired regions.

You can run disaster-recovery drills with on-demand test failovers, without affecting your production workloads or ongoing replication. Create recovery plans to orchestrate failover and fallback of the entire application running on multiple VMs. The recovery plan feature is integrated with Azure automation runbooks.

You can get started by [replicating your virtual machines](#).

## Managed snapshots

In development and test environments, snapshots provide a quick and simple option for backing up VMs that use managed disks. A managed snapshot is a full, read-only copy of a managed disk. Snapshots exist independently of their source disks.

Snapshots can be used to create new managed disks when a VM is rebuilt. They're billed based on the used portion of the disk. For example, if you create a snapshot of a managed disk with provisioned capacity of 64 GB and actual used data size of 10 GB, snapshot will be billed only for the used data size of 10 GB.

For more information on creating snapshots, see:

- [Create copy of VHD stored as a Managed Disk](#)

## Virtual machine restore points

At this time, you can use Azure REST APIs to back up and restore your VMs. This approach is most often used by independent software vendor (ISVs) or organizations with a relatively small number of VMs to manage.

You can use the API to create a VM restore point collection. The restore point collection itself contains individual restore points for specific VMs. Each restore point stores a VM's configuration and a snapshot for each attached managed disk. To save space and costs, you can exclude any disk from your VM restore points.

Once created, VM restore points can then be used to restore individual disks. To restore a VM, restore all relevant disks and attach them to a new VM.

Learn more about [working with VM restore points](#) and the [restore point collections API](#).

## Next steps

You can try out Azure Backup by following the [Azure Backup quickstart](#).

# What if an Azure service disruption impacts Azure VMs

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

At Microsoft, we work hard to make sure that our services are always available to you when you need them. Forces beyond our control sometimes impact us in ways that cause unplanned service disruptions.

Microsoft provides a Service Level Agreement (SLA) for its services as a commitment for uptime and connectivity. The SLA for individual Azure services can be found at [Azure Service Level Agreements](#).

Azure already has many built-in platform features that support highly available applications. For more about these services, read [Disaster recovery and high availability for Azure applications](#).

This article covers a true disaster recovery scenario, when a whole region experiences an outage due to major natural disaster or widespread service interruption. These are rare occurrences, but you must prepare for the possibility that there is an outage of an entire region. If an entire region experiences a service disruption, the locally redundant copies of your data would temporarily be unavailable. If you have enabled geo-replication, three additional copies of your Azure Storage blobs and tables are stored in a different region. In the event of a complete regional outage or a disaster in which the primary region is not recoverable, Azure remaps all of the DNS entries to the geo-replicated region.

To help you handle these rare occurrences, we provide the following guidance for Azure virtual machines in the case of a service disruption of the entire region where your Azure virtual machine application is deployed.

## Option 1: Initiate a failover by using Azure Site Recovery

You can configure Azure Site Recovery for your VMs so that you can recover your application with a single click in matter of minutes. You can replicate to Azure region of your choice and not restricted to paired regions. You can get started by [replicating your virtual machines](#). You can [create a recovery plan](#) so that you can automate the entire failover process for your application. You can [test your failovers](#) beforehand without impacting production application or the ongoing replication. In the event of a primary region disruption, you just [initiate a failover](#) and bring your application in target region.

## Option 2: Wait for recovery

In this case, no action on your part is required. Know that we are working diligently to restore service availability. You can see the current service status on our [Azure Service Health Dashboard](#).

This is the best option if you have not set up Azure Site Recovery, read-access geo-redundant storage, or geo-redundant storage prior to the disruption. If you have set up geo-redundant storage or read-access geo-redundant storage for the storage account where your VM virtual hard drives (VHDs) are stored, you can look to recover the base image VHD and try to provision a new VM from it. This is not a preferred option because there are no guarantees of synchronization of data. Consequently, this option is not guaranteed to work.

**NOTE**

Be aware that you do not have any control over this process, and it will only occur for region-wide service disruptions. Because of this, you must also rely on other application-specific backup strategies to achieve the highest level of availability. For more information, see the section on [Data strategies for disaster recovery](#).

## Next steps

- Start [protecting your applications running on Azure virtual machines](#) using Azure Site Recovery
- To learn more about how to implement a disaster recovery and high availability strategy, see [Disaster recovery and high availability for Azure applications](#).
- To develop a detailed technical understanding of a cloud platform's capabilities, see [Azure resiliency technical guidance](#).
- If the instructions are not clear, or if you would like Microsoft to do the operations on your behalf, contact [Customer Support](#).

# An overview of Azure VM backup

9/21/2022 • 12 minutes to read • [Edit Online](#)

This article describes how the [Azure Backup service](#) backs up Azure virtual machines (VMs).

Azure Backup provides independent and isolated backups to guard against unintended destruction of the data on your VMs. Backups are stored in a Recovery Services vault with built-in management of recovery points. Configuration and scaling are simple, backups are optimized, and you can easily restore as needed.

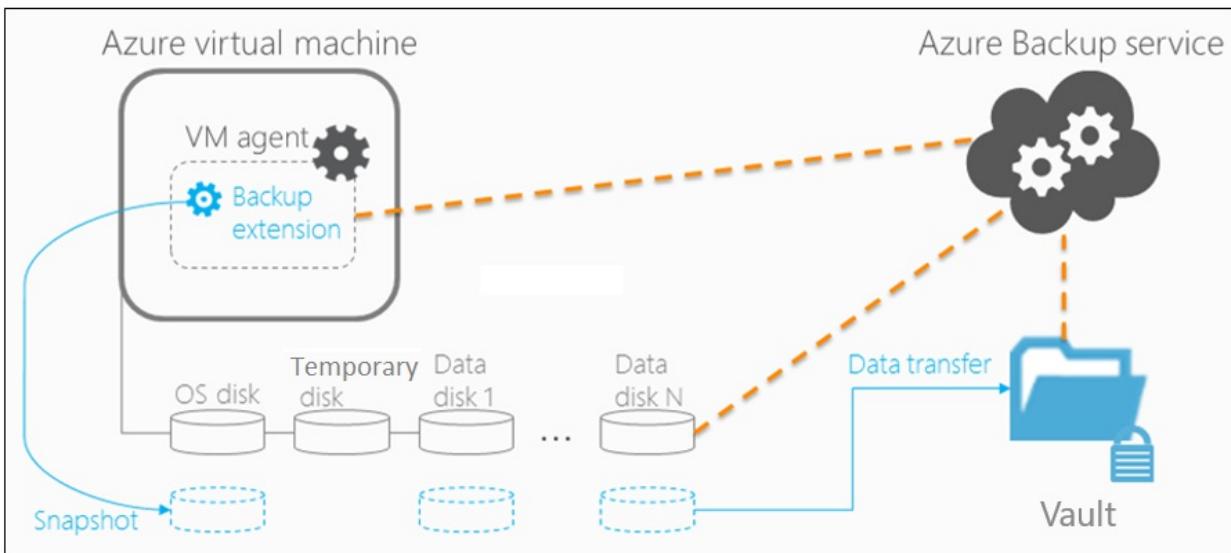
As part of the backup process, a [snapshot is taken](#), and the data is transferred to the Recovery Services vault with no impact on production workloads. The snapshot provides different levels of consistency, as described [here](#).

Azure Backup also has specialized offerings for database workloads like [SQL Server](#) and [SAP HANA](#) that are workload-aware, offer 15 minute RPO (recovery point objective), and allow backup and restore of individual databases.

## Backup process

Here's how Azure Backup completes a backup for Azure VMs:

1. For Azure VMs that are selected for backup, Azure Backup starts a backup job according to the backup schedule you specify.
2. During the first backup, a backup extension is installed on the VM if the VM is running.
  - For Windows VMs, the [VMSnapshot extension](#) is installed.
  - For Linux VMs, the [VMSnapshotLinux extension](#) is installed.
3. For Windows VMs that are running, Backup coordinates with Windows Volume Shadow Copy Service (VSS) to take an app-consistent snapshot of the VM.
  - By default, Backup takes full VSS backups.
  - If Backup can't take an app-consistent snapshot, then it takes a file-consistent snapshot of the underlying storage (because no application writes occur while the VM is stopped).
4. For Linux VMs, Backup takes a file-consistent backup. For app-consistent snapshots, you need to manually customize pre/post scripts.
5. After Backup takes the snapshot, it transfers the data to the vault.
  - The backup is optimized by backing up each VM disk in parallel.
  - For each disk that's being backed up, Azure Backup reads the blocks on the disk and identifies and transfers only the data blocks that changed (the delta) since the previous backup.
  - Snapshot data might not be immediately copied to the vault. It might take some hours at peak times. Total backup time for a VM will be less than 24 hours for daily backup policies.
6. Changes made to a Windows VM after Azure Backup is enabled on it are:
  - Microsoft Visual C++ 2013 Redistributable(x64) - 12.0.40660 is installed in the VM
  - Startup type of Volume Shadow Copy service (VSS) changed to automatic from manual
  - IaaSVmProvider Windows service is added
7. When the data transfer is complete, the snapshot is removed, and a recovery point is created.



## Encryption of Azure VM backups

When you back up Azure VMs with Azure Backup, VMs are encrypted at rest with Storage Service Encryption (SSE). Azure Backup can also back up Azure VMs that are encrypted by using Azure Disk Encryption.

ENCRYPTION	DETAILS	SUPPORT
SSE	<p>With SSE, Azure Storage provides encryption at rest by automatically encrypting data before storing it. Azure Storage also decrypts data before retrieving it. Azure Backup supports backups of VMs with two types of Storage Service Encryption:</p> <ul style="list-style-type: none"> <li><b>SSE with platform-managed keys:</b> This encryption is by default for all disks in your VMs. See more <a href="#">here</a>.</li> <li><b>SSE with customer-managed keys.</b> With CMK, you manage the keys used to encrypt the disks. See more <a href="#">here</a>.</li> </ul>	Azure Backup uses SSE for at-rest encryption of Azure VMs.
Azure Disk Encryption	<p>Azure Disk Encryption encrypts both OS and data disks for Azure VMs.</p> <p>Azure Disk Encryption integrates with BitLocker encryption keys (BEKs), which are safeguarded in a key vault as secrets. Azure Disk Encryption also integrates with Azure Key Vault key encryption keys (KEKs).</p>	<p>Azure Backup supports backup of managed and unmanaged Azure VMs encrypted with BEKs only, or with BEKs together with KEKs.</p> <p>Both BEKs and KEKs are backed up and encrypted.</p> <p>Because KEKs and BEKs are backed up, users with the necessary permissions can restore keys and secrets back to the key vault if needed. These users can also recover the encrypted VM.</p> <p>Encrypted keys and secrets can't be read by unauthorized users or by Azure.</p>

For managed and unmanaged Azure VMs, Backup supports both VMs encrypted with BEKs only or VMs encrypted with BEKs together with KEKs.

The backed-up BEKs (secrets) and KEKs (keys) are encrypted. They can be read and used only when they're

restored back to the key vault by authorized users. Neither unauthorized users, or Azure, can read or use backed-up keys or secrets.

BEKs are also backed up. So, if the BEKs are lost, authorized users can restore the BEKs to the key vault and recover the encrypted VMs. Only users with the necessary level of permissions can back up and restore encrypted VMs or keys and secrets.

## Snapshot creation

Azure Backup takes snapshots according to the backup schedule.

- **Windows VMs:** For Windows VMs, the Backup service coordinates with VSS to take an app-consistent snapshot of the VM disks. By default, Azure Backup takes a full VSS backup (it truncates the logs of application such as SQL Server at the time of backup to get application level consistent backup). If you're using a SQL Server database on Azure VM backup, then you can modify the setting to take a VSS Copy backup (to preserve logs). For more information, see [this article](#).
- **Linux VMs:** To take app-consistent snapshots of Linux VMs, use the Linux pre-script and post-script framework to write your own custom scripts to ensure consistency.
  - Azure Backup invokes only the pre/post scripts written by you.
  - If the pre-scripts and post-scripts execute successfully, Azure Backup marks the recovery point as application-consistent. However, when you're using custom scripts, you're ultimately responsible for the application consistency.
  - [Learn more](#) about how to configure scripts.

## Snapshot consistency

The following table explains the different types of snapshot consistency:

SNAPSHOT	DETAILS	RECOVERY	CONSIDERATION
Application-consistent	App-consistent backups capture memory content and pending I/O operations. App-consistent snapshots use a VSS writer (or pre/post scripts for Linux) to ensure the consistency of the app data before a backup occurs.	When you're recovering a VM with an app-consistent snapshot, the VM boots up. There's no data corruption or loss. The apps start in a consistent state.	Windows: All VSS writers succeeded  Linux: Pre/post scripts are configured and succeeded
File-system consistent	File-system consistent backups provide consistency by taking a snapshot of all files at the same time.	When you're recovering a VM with a file-system consistent snapshot, the VM boots up. There's no data corruption or loss. Apps need to implement their own "fix-up" mechanism to make sure that restored data is consistent.	Windows: Some VSS writers failed  Linux: Default (if pre/post scripts aren't configured or failed)

Snapshot	Details	Recovery	Consideration
Crash-consistent	Crash-consistent snapshots typically occur if an Azure VM shuts down at the time of backup. Only the data that already exists on the disk at the time of backup is captured and backed up.	Starts with the VM boot process followed by a disk check to fix corruption errors. Any in-memory data or write operations that weren't transferred to disk before the crash are lost. Apps implement their own data verification. For example, a database app can use its transaction log for verification. If the transaction log has entries that aren't in the database, the database software rolls transactions back until the data is consistent.	VM is in shutdown (stopped/ deallocated) state.

#### NOTE

If the provisioning state is **succeeded**, Azure Backup takes file-system consistent backups. If the provisioning state is **unavailable** or **failed**, crash-consistent backups are taken. If the provisioning state is **creating** or **deleting**, that means Azure Backup is retrying the operations.

## Backup and restore considerations

Consideration	Details
Disk	Backup of VM disks is parallel. For example, if a VM has four disks, the Backup service attempts to back up all four disks in parallel. Backup is incremental (only changed data).
Scheduling	To reduce backup traffic, back up different VMs at different times of the day and make sure the times don't overlap. Backing up VMs at the same time causes traffic jams.
Preparing backups	Keep in mind the time needed to prepare the backup. The preparation time includes installing or updating the backup extension and triggering a snapshot according to the backup schedule.

CONSIDERATION	DETAILS
Data transfer	<p>Consider the time needed for Azure Backup to identify the incremental changes from the previous backup.</p> <p>In an incremental backup, Azure Backup determines the changes by calculating the checksum of the block. If a block is changed, it's marked for transfer to the vault. The service analyzes the identified blocks to attempt to further minimize the amount of data to transfer. After evaluating all the changed blocks, Azure Backup transfers the changes to the vault.</p> <p>There might be a lag between taking the snapshot and copying it to vault. At peak times, it can take up to eight hours for the snapshots to be transferred to the vault. The backup time for a VM will be less than 24 hours for the daily backup.</p>
Initial backup	Although the total backup time for incremental backups is less than 24 hours, that might not be the case for the first backup. The time needed for the initial backup will depend on the size of the data and when the backup is processed.
Restore queue	Azure Backup processes restore jobs from multiple storage accounts at the same time, and it puts restore requests in a queue.
Restore copy	<p>During the restore process, data is copied from the vault to the storage account.</p> <p>The total restore time depends on the I/O operations per second (IOPS) and the throughput of the storage account.</p> <p>To reduce the copy time, select a storage account that isn't loaded with other application writes and reads.</p>

## Backup performance

These common scenarios can affect the total backup time:

- **Adding a new disk to a protected Azure VM:** If a VM is undergoing incremental backup and a new disk is added, the backup time will increase. The total backup time might last more than 24 hours because of initial replication of the new disk, along with delta replication of existing disks.
- **Fragmented disks:** Backup operations are faster when disk changes are contiguous. If changes are spread out and fragmented across a disk, backup will be slower.
- **Disk churn:** If protected disks that are undergoing incremental backup have a daily churn of more than 200 GB, backup can take a long time (more than eight hours) to complete.
- **Backup versions:** The latest version of Backup (known as the Instant Restore version) uses a more optimized process than checksum comparison for identifying changes. But if you're using Instant Restore and have deleted a backup snapshot, the backup switches to checksum comparison. In this case, the backup operation will exceed 24 hours (or fail).

## Restore performance

These common scenarios can affect the total restore time:

- The total restore time depends on the Input/output operations per second (IOPS) and the throughput of the storage account.
- The total restore time can be affected if the target storage account is loaded with other application read and

write operations. To improve restore operation, select a storage account that isn't loaded with other application data.

## Best practices

When you're configuring VM backups, we suggest following these practices:

- Modify the default schedule times that are set in a policy. For example, if the default time in the policy is 12:00 AM, increment the timing by several minutes so that resources are optimally used.
- If you're restoring VMs from a single vault, we highly recommend that you use different [general-purpose v2 storage accounts](#) to ensure that the target storage account doesn't get throttled. For example, each VM must have a different storage account. For example, if 10 VMs are restored, use 10 different storage accounts.
- For backup of VMs that are using premium storage with Instant Restore, we recommend allocating 50% free space of the total allocated storage space, which is required **only** for the first backup. The 50% free space isn't a requirement for backups after the first backup is complete
- The limit on the number of disks per storage account is relative to how heavily the disks are being accessed by applications that are running on an infrastructure as a service (IaaS) VM. As a general practice, if 5 to 10 disks or more are present on a single storage account, balance the load by moving some disks to separate storage accounts.
- To restore VMs with managed disks using PowerShell, provide the additional parameter **TargetResourceGroupName** to specify the resource group to which managed disks will be restored, [Learn more here](#).

## Backup costs

Azure VMs backed up with Azure Backup are subject to [Azure Backup pricing](#).

Billing doesn't start until the first successful backup finishes. At this point, the billing for both storage and protected VMs begins. Billing continues as long as any backup data for the VM is stored in a vault. If you stop protection for a VM, but backup data for the VM exists in a vault, billing continues.

Billing for a specified VM stops only if the protection is stopped and all backup data is deleted. When protection stops and there are no active backup jobs, the size of the last successful VM backup becomes the protected instance size used for the monthly bill.

The protected-instance size calculation is based on the *actual* size of the VM. The VM's size is the sum of all the data in the VM, excluding the temporary storage. Pricing is based on the actual data that's stored on the data disks, not on the maximum supported size for each data disk that's attached to the VM.

Similarly, the backup storage bill is based on the amount of data that's stored in Azure Backup, which is the sum of the actual data in each recovery point.

For example, take an A2-Standard-sized VM that has two additional data disks with a maximum size of 32 TB each. The following table shows the actual data stored on each of these disks:

DISK	MAX SIZE	ACTUAL DATA PRESENT
OS disk	32 TB	17 GB
Local/temporary disk	135 GB	5 GB (not included for backup)
Data disk 1	32 TB	30 GB
Data disk 2	32 TB	0 GB

The actual size of the VM in this case is 17 GB + 30 GB + 0 GB = 47 GB. This protected-instance size (47 GB) becomes the basis for the monthly bill. As the amount of data in the VM grows, the protected-instance size used for billing changes to match.

## Next steps

- [Prepare for Azure VM backup.](#)

# Back up a virtual machine in Azure with the Azure CLI

9/21/2022 • 6 minutes to read • [Edit Online](#)

The Azure CLI is used to create and manage Azure resources from the command line or in scripts. You can protect your data by taking backups at regular intervals. Azure Backup creates recovery points that can be stored in geo-redundant recovery vaults. This article details how to back up a virtual machine (VM) in Azure with the Azure CLI. You can also perform these steps with [Azure PowerShell](#) or in the [Azure portal](#).

This quickstart enables backup on an existing Azure VM. If you need to create a VM, you can [create a VM with the Azure CLI](#).

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.
- This quickstart requires version 2.0.18 or later of the Azure CLI. If using Azure Cloud Shell, the latest version is already installed.

## Create a Recovery Services vault

A Recovery Services vault is a logical container that stores the backup data for each protected resource, such as Azure VMs. When the backup job for a protected resource runs, it creates a recovery point inside the Recovery Services vault. You can then use one of these recovery points to restore data to a given point in time.

Create a Recovery Services vault with `az backup vault create`. Specify the same resource group and location as the VM you wish to protect. If you used the [VM quickstart](#), then you created:

- a resource group named *myResourceGroup*,
- a VM named *myVM*,
- resources in the *eastus* location.

```
az backup vault create --resource-group myResourceGroup \
--name myRecoveryServicesVault \
--location eastus
```

By default, the Recovery Services vault is set for Geo-Redundant storage. Geo-Redundant storage ensures your backup data is replicated to a secondary Azure region that's hundreds of miles away from the primary region. If the storage redundancy setting needs to be modified, use [az backup vault backup-properties set cmdlet](#).

```
az backup vault backup-properties set \
--name myRecoveryServicesVault \
--resource-group myResourceGroup \
--backup-storage-redundancy "LocallyRedundant/GeoRedundant"
```

## Enable backup for an Azure VM

Create a protection policy to define: when a backup job runs, and how long the recovery points are stored. The default protection policy runs a backup job each day and retains recovery points for 30 days. You can use these default policy values to quickly protect your VM. To enable backup protection for a VM, use [az backup protection enable-for-vm](#). Specify the resource group and VM to protect, then the policy to use:

```
az backup protection enable-for-vm \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--vm myVM \
--policy-name DefaultPolicy
```

### NOTE

If the VM isn't in the same resource group as that of the vault, then myResourceGroup refers to the resource group where vault was created. Instead of VM name, provide the VM ID as indicated below.

```
az backup protection enable-for-vm \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--vm $(az vm show -g VMResourceGroup -n MyVm --query id | tr -d '') \
--policy-name DefaultPolicy
```

### IMPORTANT

While using CLI to enable backup for multiple VMs at once, ensure that a single policy doesn't have more than 100 VMs associated with it. This is a [recommended best practice](#). Currently, the PowerShell client doesn't explicitly block if there are more than 100 VMs, but the check is planned to be added in the future.

## Prerequisites to backup encrypted VMs

To enable protection on encrypted VMs (encrypted using BEK and KEK), you must provide the Azure Backup service permission to read keys and secrets from the key vault. To do so, set a *keyvault* access policy with the required permissions, as demonstrated below:

```

# Enter the name of the resource group where the key vault is located on this variable
AZ_KEYVAULT_RGROUP=TestKeyVaultRG

# Enter the name of the key vault on this variable
AZ_KEYVAULT_NAME=TestKeyVault

# Get the object id for the Backup Management Service on your subscription
AZ_ABM_OBJECT_ID=$( az ad sp list --display-name "Backup Management Service" --query '[].objectId' -o tsv --only-show-errors )

# This command will grant the permissions required by the Backup Management Service to access the key vault
az keyvault set-policy --key-permissions get list backup --secret-permissions get list backup \
--resource-group $AZ_KEYVAULT_RGROUP --name $AZ_KEYVAULT_NAME --object-id $AZ_ABM_OBJECT_ID

```

## Start a backup job

To start a backup now rather than wait for the default policy to run the job at the scheduled time, use [az backup protection backup-now](#). This first backup job creates a full recovery point. Each backup job after this initial backup creates incremental recovery points. Incremental recovery points are storage and time-efficient, as they only transfer changes made since the last backup.

The following parameters are used to back up the VM:

- `--container-name` is the name of your VM
- `--item-name` is the name of your VM
- `--retain-until` value should be set to the last available date, in UTC time format (**dd-mm-yyyy**), that you wish the recovery point to be available

The following example backs up the VM named *myVM* and sets the expiration of the recovery point to October 18, 2017:

```

az backup protection backup-now \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--container-name myVM \
--item-name myVM \
--backup-management-type AzureIaaSVM
--retain-until 18-10-2017

```

## Monitor the backup job

To monitor the status of backup jobs, use [az backup job list](#):

```

az backup job list \
--resource-group myResourceGroup \
--vault-name myRecoveryServicesVault \
--output table

```

The output is similar to the following example, which shows the backup job is *InProgress*.

Name	Operation	Status	Item Name	Start Time UTC	Duration
a0a8e5e6	Backup	InProgress	myvm	2017-09-19T03:09:21	0:00:48.718366
fe5d0414	ConfigureBackup	Completed	myvm	2017-09-19T03:03:57	0:00:31.191807

When the *Status* of the backup job reports *Completed*, your VM is protected with Recovery Services and has a

full recovery point stored.

## Clean up deployment

When no longer needed, you can disable protection on the VM, remove the restore points and Recovery Services vault, then delete the resource group and associated VM resources. If you used an existing VM, you can skip the final `az group delete` command to leave the resource group and VM in place.

If you want to try a Backup tutorial that explains how to restore data for your VM, go to [Next steps](#).

```
az backup protection disable \
    --resource-group myResourceGroup \
    --vault-name myRecoveryServicesVault \
    --container-name myVM \
    --item-name myVM \
    --backup-management-type AzureIaaSVM
    --delete-backup-data true
az backup vault delete \
    --resource-group myResourceGroup \
    --name myRecoveryServicesVault \
az group delete --name myResourceGroup
```

## Next steps

In this quickstart, you created a Recovery Services vault, enabled protection on a VM, and created the initial recovery point. To learn more about Azure Backup and Recovery Services, continue to the tutorials.

[Back up multiple Azure VMs](#)

# Back up a virtual machine in Azure with PowerShell

9/21/2022 • 4 minutes to read • [Edit Online](#)

The [Azure PowerShell AZ](#) module is used to create and manage Azure resources from the command line or in scripts.

[Azure Backup](#) backs up on-premises machines and apps, and Azure VMs. This article shows you how to back up an Azure VM with the AZ module. Alternatively, you can back up a VM using the [Azure CLI](#), or in the [Azure portal](#).

This quickstart enables backup on an existing Azure VM. If you need to create a VM, you can [create a VM with Azure PowerShell](#).

This quickstart requires the Azure PowerShell AZ module version 1.0.0 or later. Run

```
Get-Module -ListAvailable Az
```

to find the version. If you need to install or upgrade, see [Install Azure PowerShell module](#).

## NOTE

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az](#).

## Sign in and register

1. Sign in to your Azure subscription with the `Connect-AzAccount` command and follow the on-screen directions.

```
Connect-AzAccount
```

2. The first time you use Azure Backup, you must register the Azure Recovery Service provider in your subscription with [Register-AzResourceProvider](#), as follows:

```
Register-AzResourceProvider -ProviderNamespace "Microsoft.RecoveryServices"
```

## Create a Recovery Services vault

A [Recovery Services vault](#) is a logical container that stores backup data for protected resources, such as Azure VMs. When a backup job runs, it creates a recovery point inside the Recovery Services vault. You can then use one of these recovery points to restore data to a given point in time.

When you create the vault:

- For the resource group and location, specify the resource group and location of the VM you want to back up.
- If you used this [sample script](#) to create the VM, the resource group is `myResourceGroup`, the VM is `*myVM`, and the resources are in the `WestEurope` region.
- Azure Backup automatically handles storage for backed up data. By default the vault uses [Geo-Redundant Storage \(GRS\)](#). Geo-redundancy ensures that backed up data is replicated to a secondary Azure region, hundreds of miles away from the primary region.

Now create a vault:

1. Use the [New-AzRecoveryServicesVault](#) to create the vault:

```
New-AzRecoveryServicesVault  
-ResourceGroupName "myResourceGroup"  
-Name "myRecoveryServicesVault"  
-Location "WestEurope"
```

2. Set the vault context with [Set-AzRecoveryServicesVaultContext](#), as follows:

```
Get-AzRecoveryServicesVault  
-Name "myRecoveryServicesVault" | Set-AzRecoveryServicesVaultContext
```

3. Change the storage redundancy configuration (LRS/GRS) of the vault with [Set-AzRecoveryServicesBackupProperty](#), as follows:

```
Get-AzRecoveryServicesVault  
-Name "myRecoveryServicesVault" | Set-AzRecoveryServicesBackupProperty -BackupStorageRedundancy  
LocallyRedundant/GeoRedundant
```

**NOTE**

Storage Redundancy can be modified only if there are no backup items protected to the vault.

## Enable backup for an Azure VM

You enable backup for an Azure VM, and specify a backup policy.

- The policy defines when backups run, and how long recovery points created by the backups should be retained.
- The default protection policy runs a backup once a day for the VM, and retains the created recovery points for 30 days. You can use this default policy to quickly protect your VM.

Enable backup as follows:

1. First, set the default policy with [Get-AzRecoveryServicesBackupProtectionPolicy](#):

```
$policy = Get-AzRecoveryServicesBackupProtectionPolicy -Name "DefaultPolicy"
```

2. Enable VM backup with [Enable-AzRecoveryServicesBackupProtection](#). Specify the policy, the resource group, and the VM name.

```
Enable-AzRecoveryServicesBackupProtection  
-ResourceGroupName "myResourceGroup"  
-Name "myVM"  
-Policy $policy
```

## Start a backup job

Backups run according to the schedule specified in the backup policy. You can also run an on-demand backup:

- The first initial backup job creates a full recovery point.

- After the initial backup, each backup job creates incremental recovery points.
- Incremental recovery points are storage and time-efficient, as they only transfer changes made since the last backup.

To run an on-demand backup, you use the [Backup-AzRecoveryServicesBackupItem](#).

- You specify a container in the vault that holds your backup data with [Get-AzRecoveryServicesBackupContainer](#).
- Each VM to back up is treated as an item. To start a backup job, you obtain information about the VM with [Get-AzRecoveryServicesBackupItem](#).

Run an on-demand backup job as follows:

1. Specify the container, obtain VM information, and run the backup.

```
$backupcontainer = Get-AzRecoveryServicesBackupContainer ` 
    -ContainerType "AzureVM" ` 
    -FriendlyName "myVM"

$item = Get-AzRecoveryServicesBackupItem ` 
    -Container $backupcontainer ` 
    -WorkloadType "AzureVM"

Backup-AzRecoveryServicesBackupItem -Item $item
```

2. You might need to wait up to 20 minutes, since the first backup job creates a full recovery point. Monitor the job as described in the next procedure.

## Monitor the backup job

1. Run [Get-AzRecoveryServicesBackupJob](#) to monitor the job status.

```
Get-AzRecoveryServicesBackupJob
```

Output is similar to the following example, which shows the job as **InProgress**:

WorkloadName	Operation	Status	StartTime	EndTime	JobID
myvm	Backup	InProgress	9/18/2017 9:38:02 PM		9f9e8f14
myvm	ConfigureBackup	Completed	9/18/2017 9:33:18 PM	9/18/2017 9:33:51 PM	fe79c739

2. When the job status is **Completed**, the VM is protected and has a full recovery point stored.

## Manage VM backups

If you want to perform more actions such as change policy, edit policy etc.. refer to the [manage VM backups section](#).

## Clean up the deployment

If you no longer need to back up the VM, you can clean it up.

- If you want to try out restoring the VM, skip the clean-up.
- If you used an existing VM, you can skip the final [Remove-AzResourceGroup](#) cmdlet to leave the resource group and VM in place.

Disable protection, remove the restore points and vault. Then delete the resource group and associated VM resources, as follows:

```
Disable-AzRecoveryServicesBackupProtection -Item $item -RemoveRecoveryPoints  
$vault = Get-AzRecoveryServicesVault -Name "myRecoveryServicesVault"  
Remove-AzRecoveryServicesVault -Vault $vault  
Remove-AzResourceGroup -Name "myResourceGroup"
```

## Next steps

In this quickstart, you created a Recovery Services vault, enabled protection on a VM, and created the initial recovery point.

- [Learn how](#) to back up VMs in the Azure portal.
- [Learn how](#) to quickly restore a VM

# Back up a virtual machine in Azure

9/21/2022 • 10 minutes to read • [Edit Online](#)

Azure backups can be created through the Azure portal. This method provides a browser-based user interface to create and configure Azure backups and all related resources. You can protect your data by taking backups at regular intervals. Azure Backup creates recovery points that can be stored in geo-redundant recovery vaults. This article details how to back up a virtual machine (VM) with the Azure portal.

This quickstart enables backup on an existing Azure VM. If you need to create a VM, you can [create a VM with the Azure portal](#).

## Sign in to Azure

Sign in to the [Azure portal](#).

### NOTE

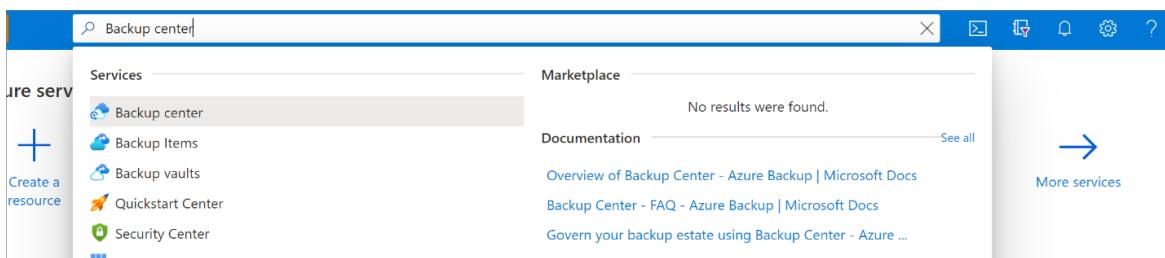
The functionality described in the following sections can also be accessed via [Backup center](#). Backup center is a single unified management experience in Azure. It enables enterprises to govern, monitor, operate, and analyze backups at scale. With this solution, you can perform most of the key backup management operations without being limited to the scope of an individual vault.

## Create a Recovery Services vault

A Recovery Services vault is a management entity that stores recovery points created over time and provides an interface to perform backup-related operations. These operations include taking on-demand backups, performing restores, and creating backup policies.

To create a Recovery Services vault:

1. Sign in to your subscription in the [Azure portal](#).
2. Search for **Backup center** in the Azure portal, and go to the **Backup Center** dashboard.



3. Select **+Vault** from the **Overview** tab.

**Backup center** Microsoft

Search (Ctrl+ /) <> + Backup ⌂ Restore + Policy + Vault ⌂ Refresh

Datasource subscription == **66 selected** Datasource resource group == All Datasource location == All Datasource type == Azure Virtual machines

**Datasource type: Azure Virtual machines**

Overview of Jobs and Backup instances

Jobs (last 24 Hours)				<a href="#">View all</a>
Operation	Failed	In progress	Completed	
Scheduled backup	64	8	246	
On-demand backup	0	0	12	
Restore	1	0	57	

Backup instances			
Azure Virtual machines			
<b>359</b>	Protection configured	<b>338</b>	
	Protection stopped	<b>16</b>	
	Soft deleted	<b>5</b>	
<b>86</b> out of 359	Backup instances with the underlying datasource not found		

#### 4. Select Recovery Services vault > Continue.

Home > Backup center > Start: Create Vault ...

A vault is an entity that stores the backups and restore points created over time. The vault also contains the backup policies that are associated with the protected virtual machines. Proceed to vault creation by selecting vault type.

Vault Type

Recovery Services vault

Supported datasources

- ✓ Azure Virtual machines
- ✓ SQL in Azure VM
- ✓ Azure Files (Azure Storage)
- ✓ SAP HANA in Azure VM
- ✓ Azure Backup Server
- ✓ Azure Backup Agent
- ✓ DPM

Backup vault

Supported datasources

- ✓ Azure Database for PostgreSQL servers (Preview)
- ✓ Azure Blobs (Azure Storage)
- ✓ Azure Disks

Learn more about Backup vault. [Click here](#) ↗

Learn more about Recovery Services vault. [Click here](#) ↗

[Continue](#) [Cancel](#)

#### 5. The Recovery Services vault dialog opens. Provide the following values:

- Subscription:** Choose the subscription to use. If you're a member of only one subscription, you'll see that name. If you're not sure which subscription to use, use the default (suggested) subscription. There are multiple choices only if your work or school account is associated with more than one Azure subscription.
- Resource group:** Use an existing resource group or create a new one. To see the list of available resource groups in your subscription, select **Use existing**, and then select a resource from the dropdown list. To create a new resource group, select **Create new** and enter the name. For more information about resource groups, see [Azure Resource Manager overview](#).
- Vault name:** Enter a friendly name to identify the vault. The name must be unique to the Azure

subscription. Specify a name that has at least 2 but not more than 50 characters. The name must start with a letter and consist only of letters, numbers, and hyphens.

- **Region:** Select the geographic region for the vault. For you to create a vault to help protect any data source, the vault *must* be in the same region as the data source.

#### IMPORTANT

If you're not sure of the location of your data source, close the dialog. Go to the list of your resources in the portal. If you have data sources in multiple regions, create a Recovery Services vault for each region. Create the vault in the first location before you create the vault for another location. There's no need to specify storage accounts to store the backup data. The Recovery Services vault and Azure Backup handle that automatically.

The screenshot shows the 'Create Recovery Services vault' wizard in the 'Basics' step. It includes fields for 'Subscription' (dropdown), 'Resource group' (dropdown with 'Create new' option), 'Vault name' (text input placeholder 'Enter the name for your vault.'), and 'Region' (dropdown set to 'East US'). Navigation buttons at the bottom are 'Review + create' (highlighted in red) and 'Next: Tags'.

6. After you provide the values, select **Review + create**.

The screenshot shows the 'Review + create' step with the 'Create' button highlighted in red. Other buttons include 'Review + create' (highlighted in blue) and 'Next: Tags'.

7. When you're ready to create the Recovery Services vault, select **Create**.

The screenshot shows the final step with the 'Create' button highlighted in red. Other buttons include 'Previous: Tags' and 'Download a template for automation'.

8. It can take a while to create the Recovery Services vault. Monitor the status notifications in the **Notifications** area at the upper-right corner of the portal. After your vault is created, it's visible in the list of Recovery Services vaults. If you don't see your vault, select **Refresh**.

The screenshot shows the 'Recovery Services vaults' list page. The 'Refresh' button in the toolbar is highlighted in red. Other buttons include '+ Add', 'Edit columns', 'Try preview', and 'Assign tags'.

# Apply a backup policy

To apply a backup policy to your Azure VMs, follow these steps:

1. Go to **Backup center** and click **+ Backup** from the **Overview** tab.

The screenshot shows the Azure Backup center interface. The left sidebar includes links for Overview, Getting started, Community, Manage (Backup instances, Backup policies, Vaults), Monitoring + reporting (Backup jobs, Backup reports), Policy and compliance (Backup compliance, Azure policies for backup, Protectable datasources), and Support + troubleshooting (New support request). The main area displays 'Jobs (last 24 Hours)' with a table showing counts for Failed, In progress, and Completed operations (Scheduled backup: 64 Failed, 8 In progress, 246 Completed; On-demand backup: 0 Failed, 0 In progress, 12 Completed; Restore: 1 Failed, 0 In progress, 57 Completed). Below this is a 'Backup instances' section for 'Azure Virtual machines', showing a total of 359 instances: 338 with protection configured, 16 stopped, and 5 soft deleted. A note indicates 86 instances have underlying datasources not found. A red box highlights the '+ Backup' button in the top navigation bar.

2. Select **Azure Virtual machines** as the **Datasource type** and select the vault you have created. Then click **Continue**.

The screenshot shows the 'Initiate: Configure Backup' step. It has three dropdown fields: 'Datasource type' set to 'Azure Virtual machines', 'Vault type' set to 'Recovery Services Vault', and 'Vault \*' with a placeholder 'Select a Vault' and a 'Select' button. A blue info box at the bottom left states: 'Selected vault is maintained in Recovery Services Vault, you will be redirected to Recovery Services Vault for configuring backup. Learn more.' The URL in the browser header is 'Home > Backup Center > Initiate: Configure Backup'.

3. Assign a Backup policy.

- The default policy backs up the VM once a day. The daily backups are retained for *30 days*. Instant recovery snapshots are retained for two days.

Home > Backup center > Start: Configure Backup >

## Configure Backup

VaultWithTag

Backup policy \* ⓘ DefaultPolicy  Create a new policy

**Policy Details**

Full Backup      **Backup Frequency**  
Daily at 6:00 PM UTC

Instant Restore  
Retain instant recovery snapshot(s) for 2 day(s)

Retention of daily backup point  
Retain backup taken every day at 6:00 PM for 30 Day(s)

**Virtual machines**

Name	Resource Group	OS Disk Only
No Virtual machines selected.		

Add

**OS Disk only backup** option allows you to backup Azure Virtual Machine with only OS disk and exclude all the data disks. You can use Selective Disk Backup feature through Powershell or CLI to include or exclude specific data disks. Know more about Selective Disk Backup feature, its limitation and pricing- [Learn more](#).

- If you don't want to use the default policy, select **Create New**, and create a custom policy as described in the next procedure.

## Select a VM to back up

Create a simple scheduled daily backup to a Recovery Services vault.

1. Under **Virtual Machines**, select Add.

**Virtual Machines**

Virtual machine name	Resource Group
No Virtual Machines Selected	

Add

2. The **Select virtual machines** pane will open. Select the VMs you want to back up using the policy. Then select **OK**.

- The selected VMs are validated.
- You can only select VMs in the same region as the vault.
- VMs can only be backed up in a single vault.

## Select virtual machines

X

Filter items ...

Virtual machine name	Resource Group
<input checked="" type="checkbox"/> myVM	myResourceGroup
<input checked="" type="checkbox"/> myVMH1	myResourceGroup
<input checked="" type="checkbox"/> myVMR1	myResourceGroup

**OK**

### NOTE

All the VMs in the same region and subscription as that of the vault are available to configure backup. When configuring backup, you can browse to the virtual machine name and its resource group, even though you don't have the required permission on those VMs. If your VM is in soft deleted state, then it won't be visible in this list. If you need to re-protect the VM, then you need to wait for the soft delete period to expire or undelete the VM from the soft deleted list. For more information, see [the soft delete for VMs article](#).

## Enable backup on a VM

A Recovery Services vault is a logical container that stores the backup data for each protected resource, such as Azure VMs. When the backup job for a protected resource runs, it creates a recovery point inside the Recovery Services vault. You can then use one of these recovery points to restore data to a given point in time.

To enable VM backup, in **Backup**, select **Enable backup**. This deploys the policy to the vault and to the VMs, and installs the backup extension on the VM agent running on the Azure VM.

After enabling backup:

- The Backup service installs the backup extension whether or not the VM is running.
- An initial backup will run in accordance with your backup schedule.
- When backups run, note that:
  - A VM that's running has the greatest chance for capturing an application-consistent recovery point.
  - However, even if the VM is turned off, it's backed up. Such a VM is known as an offline VM. In this case, the recovery point will be crash-consistent.
- Explicit outbound connectivity isn't required to allow backup of Azure VMs.

### Create a custom policy

If you selected to create a new backup policy, fill in the policy settings.

1. In **Policy name**, specify a meaningful name.
2. In **Backup schedule**, specify when backups should be taken. You can take daily or weekly backups for Azure VMs.

3. In **Instant Restore**, specify how long you want to retain snapshots locally for instant restore.
  - When you restore, backed up VM disks are copied from storage, across the network to the recovery storage location. With instant restore, you can leverage locally stored snapshots taken during a backup job, without waiting for backup data to be transferred to the vault.
  - You can retain snapshots for instant restore for between one to five days. The default value is two days.
4. In **Retention range**, specify how long you want to keep your daily or weekly backup points.
5. In **Retention of monthly backup point** and **Retention of yearly backup point**, specify whether you want to keep a monthly or yearly backup of your daily or weekly backups.
6. Select **OK** to save the policy.

**NOTE**

To store the restore point collection (RPC), the Backup service creates a separate resource group (RG). This RG is different than RG of the VM. [Learn more](#).

## Backup policy

X

Policy name \* ⓘ

Backup schedule

Frequency \*

 Daily

Time \*

 11:00 AM

Timezone \*

 (UTC) Coordinated Universal Time

Instant Restore ⓘ

Retain instant recovery snapshot(s) for

 2

Day(s) ⓘ

Retention range

Retention of daily backup point.

At

For

 11:00 AM 180

Day(s)

Retention of weekly backup point.

On \*

At

For

 Sunday 11:00 AM 12

Week(s)

Retention of monthly backup point.

Week Based  Day Based

On \*

Day \*

At

For

 First Sunday 11:00 AM 60

Month(s)

Retention of yearly backup point.

Week Based  Day Based

In \*

On \*

Day \*

At

For

 January First Sunday 11:00 AM 10

Year(s)



Azure Backup service creates a separate resource group to store the instant recovery points of managed virtual machines. The default naming format of resource group created by Azure Backup service is AzureBackupRG\_{Geo}\_{n}. It is optional to customize the name as per your requirement. [Learn More](#)

Azure Backup Resource Group (Optional) ⓘ

Enter the name

n

Suffix (Optional)

OK

### NOTE

Azure Backup doesn't support automatic clock adjustment for daylight-saving changes for Azure VM backups. As time changes occur, modify backup policies manually as required.

Start a backup job

The initial backup will run in accordance with the schedule, but you can run it immediately as follows:

1. Go to **Backup center** and select the **Backup Instances** menu item.
2. Select **Azure Virtual machines** as the **Datasource type**. Then search for the VM that you have configured for backup.
3. Right-click the relevant row or select the more icon (...), and then click **Backup Now**.
4. In **Backup Now**, use the calendar control to select the last day that the recovery point should be retained. Then select **OK**.
5. Monitor the portal notifications. To monitor the job progress, go to **Backup center > Backup Jobs** and filter the list for **In progress** jobs. Depending on the size of your VM, creating the initial backup may take a while.

## Monitor the backup job

The Backup job details for each VM backup consist of two phases, the **Snapshot** phase followed by the **Transfer data to vault** phase.

The snapshot phase guarantees the availability of a recovery point stored along with the disks for **Instant Restores** and are available for a maximum of five days depending on the snapshot retention configured by the user. Transfer data to vault creates a recovery point in the vault for long-term retention. Transfer data to vault only starts after the snapshot phase is completed.

Backup instance	Datasource subs...	Datasource reso...	Datasource locat...	Operation	Status	Vault	Start time	Duration	...
CH1-JBOXVM00	Contoso Hotels Tena...	CH1-OpsRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:24:04 AM	02:21:14	...
CH1-DCVM01	Contoso Hotels Tena...	CH1-InfraRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:22:23 AM	02:21:14	...
CH1-AppBEVM01	Contoso Hotels Tena...	CH1-RetailRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:21:46 AM	02:21:17	...
CH1-SQLVM01	Contoso Hotels Tena...	CH1-RetailRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:20:21 AM	02:51:18	...
CH1-AppBEVM00	Contoso Hotels Tena...	CH1-RetailRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:17:36 AM	02:26:17	...
CH1-SQLVM00	Contoso Hotels Tena...	CH1-RetailRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:15:45 AM	02:51:18	...
CH1-DCVM00	Contoso Hotels Tena...	CH1-InfraRG-Pri	East US	Scheduled Backup	Completed	CH1-RV-Pri	3/8/2021, 7:15:23 AM	02:51:14	...
CH1-JBOXVM10	Contoso Hotels Tena...	CH1-OpsRG-Sec	West US 2	Scheduled Backup	Completed	CH1-RV-Sec	3/8/2021, 7:07:28 AM	02:21:12	...
CH1-SQLVM12	Contoso Hotels Tena...	CH1-RetailRG-Sec	West US 2	Scheduled Backup	Completed	CH1-RV-Sec	3/8/2021, 7:07:00 AM	02:31:17	...
CH1-DCVM11	Contoso Hotels Tena...	CH1-InfraRG-Sec	West US 2	Scheduled Backup	Completed	CH1-RV-Sec	3/8/2021, 7:04:54 AM	02:21:13	...
CH1-DCVM10	Contoso Hotels Tena...	CH1-InfraRG-Sec	West US 2	Scheduled Backup	Completed	CH1-RV-Sec	3/8/2021, 7:01:17 AM	02:21:14	...

There are two **Sub Tasks** running at the backend, one for front-end backup job that can be checked from the **Backup Job** details pane as given below:

Name	Status
Take Snapshot	Completed
Transfer data to vault	Completed

The **Transfer data to vault** phase can take multiple days to complete depending on the size of the disks, churn

per disk and several other factors.

Job status can vary depending on the following scenarios:

SNAPSHOT	TRANSFER DATA TO VAULT	JOB STATUS
Completed	In progress	In progress
Completed	Skipped	Completed
Completed	Completed	Completed
Completed	Failed	Completed with warning
Failed	Failed	Failed

Now with this capability, for the same VM, two backups can run in parallel, but in either phase (snapshot, transfer data to vault) only one sub task can be running. So in scenarios where a backup job in progress resulted in the next day's backup to fail, it will be avoided with this decoupling functionality. Subsequent days' backups can have the snapshot completed, while **Transfer data to vault** is skipped if an earlier day's backup job is in progress state. The incremental recovery point created in the vault will capture all the churn from the most recent recovery point created in the vault. There's no cost impact on the user.

## Optional steps

### Install the VM agent

Azure Backup backs up Azure VMs by installing an extension to the Azure VM agent running on the machine. If your VM was created from an Azure Marketplace image, the agent is installed and running. If you create a custom VM, or you migrate an on-premises machine, you might need to install the agent manually, as summarized in the table.

VM	DETAILS
Windows	<ol style="list-style-type: none"><li>1. <a href="#">Download and install</a> the agent MSI file.</li><li>2. Install with admin permissions on the machine.</li><li>3. Verify the installation. In <i>C:\WindowsAzure\Packages</i> on the VM, right-click <b>WaAppAgent.exe</b> &gt; <b>Properties</b>. On the <b>Details</b> tab, <b>Product Version</b> should be 2.6.1198.718 or higher.</li></ol> <p>If you're updating the agent, make sure that no backup operations are running, and <a href="#">reinstall the agent</a>.</p>
Linux	<p>Install by using an RPM or a DEB package from your distribution's package repository. This is the preferred method for installing and upgrading the Azure Linux agent. All the <a href="#">endorsed distribution providers</a> integrate the Azure Linux agent package into their images and repositories. The agent is available on <a href="#">GitHub</a>, but we don't recommend installing from there.</p> <p>If you're updating the agent, make sure no backup operations are running, and update the binaries.</p>

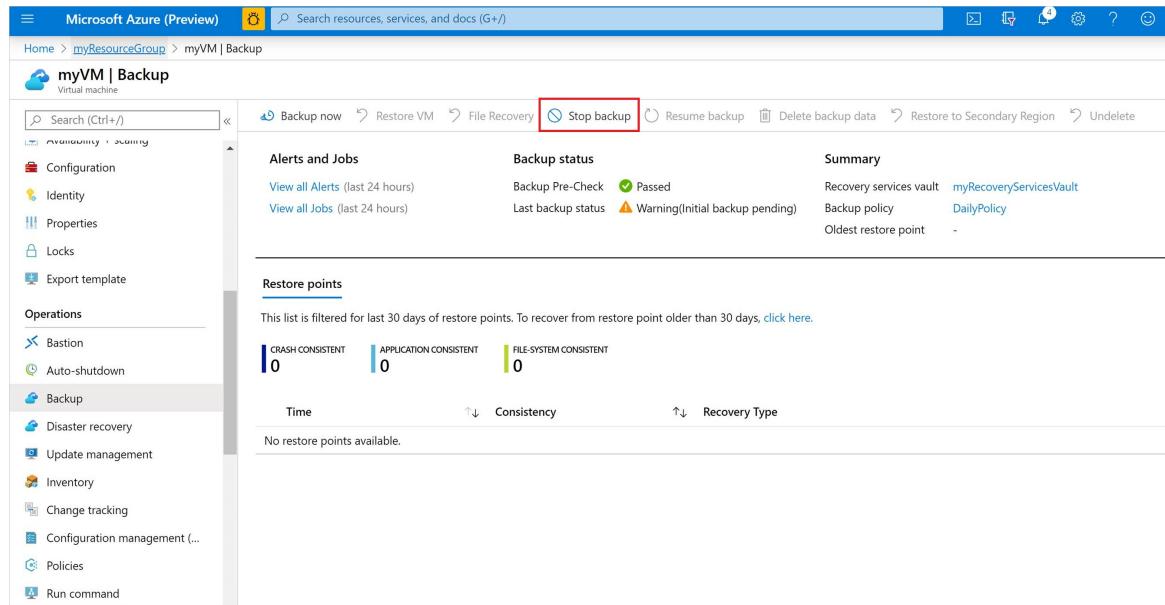
# Clean up deployment

When no longer needed, you can disable protection on the VM, remove the restore points and Recovery Services vault, then delete the resource group and associated VM resources

If you're going to continue on to a Backup tutorial that explains how to restore data for your VM, skip the steps in this section and go to [Next steps](#).

1. Select the **Backup** option for your VM.

2. Choose **Stop backup**.



The screenshot shows the Azure portal interface for a virtual machine named 'myVM'. On the left, there's a navigation menu with options like Configuration, Identity, Properties, Locks, Export template, Operations, Bastion, Auto-shutdown, Backup (which is selected), Disaster recovery, Update management, Inventory, Change tracking, Configuration management, Policies, and Run command. The main content area is titled 'myVM | Backup'. It has sections for 'Alerts and Jobs', 'Backup status' (showing a green 'Passed' status), 'Summary' (Recovery services vault: myRecoveryServicesVault, Backup policy: DailyPolicy), and 'Restore points' (which is currently empty). At the top right, there are several buttons: 'Backup now', 'Restore VM', 'File Recovery', 'Stop backup' (highlighted with a red box), 'Resume backup', 'Delete backup data', 'Restore to Secondary Region', and 'Undelete'.

3. Select **Delete Backup Data** from the drop-down menu.

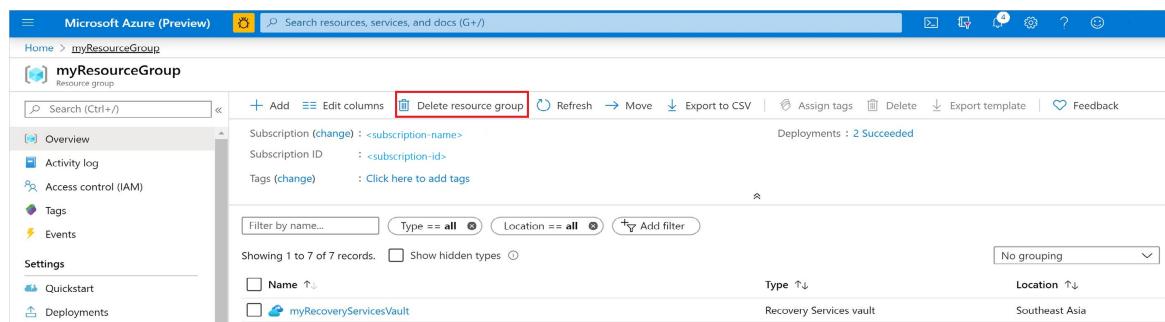
4. In the **Type the name of the Backup item** dialog, enter your VM name, such as *myVM*. Select **Stop Backup**.

Once the VM backup has been stopped and recovery points removed, you can delete the resource group. If you used an existing VM, you may wish to leave the resource group and VM in place.

5. In the menu on the left, select **Resource groups**.

6. From the list, choose your resource group. If you used the sample VM quickstart commands, the resource group is named *myResourceGroup*.

7. Select **Delete resource group**. To confirm, enter the resource group name, then select **Delete**.



The screenshot shows the Azure portal interface for a resource group named 'myResourceGroup'. On the left, there's a navigation menu with options like Overview, Activity log, Access control (IAM), Tags, Events, Settings, Quickstart, and Deployments. The main content area shows details for the resource group, including Subscription (change), Subscription ID, and Tags (change). It also lists deployments (2 Succeeded). Below this is a table showing resource group details: Name (myResourceGroup), Type (Recovery Services vault), Location (Southeast Asia). At the top right, there are buttons for '+ Add', 'Edit columns', 'Delete resource group' (highlighted with a red box), Refresh, Move, Export to CSV, Assign tags, Delete, Export template, and Feedback.

## Next steps

In this quickstart, you created a Recovery Services vault, enabled protection on a VM, and created the initial recovery point. To learn more about Azure Backup and Recovery Services, continue to the tutorials.

[Back up multiple Azure VMs](#)

# Back up a virtual machine in Azure with an ARM template

9/21/2022 • 5 minutes to read • [Edit Online](#)

Azure Backup backs up on-premises machines and apps, and Azure VMs. This article shows you how to back up an Azure VM with an Azure Resource Manager template (ARM template) and Azure PowerShell. This quickstart focuses on the process of deploying an ARM template to create a Recovery Services vault. For more information on developing ARM templates, see the [Azure Resource Manager documentation](#) and the [template reference](#).

An [ARM template](#) is a JavaScript Object Notation (JSON) file that defines the infrastructure and configuration for your project. The template uses declarative syntax. In declarative syntax, you describe your intended deployment without writing the sequence of programming commands to create the deployment.

A [Recovery Services vault](#) is a logical container that stores backup data for protected resources, such as Azure VMs. When a backup job runs, it creates a recovery point inside the Recovery Services vault. You can then use one of these recovery points to restore data to a given point in time. Alternatively, you can back up a VM using [Azure PowerShell](#), the [Azure CLI](#), or in the [Azure portal](#).

If your environment meets the prerequisites and you're familiar with using ARM templates, select the **Deploy to Azure** button. The template will open in the Azure portal.



## Review the template

The template used in this quickstart is from [Azure quickstart Templates](#). This template allows you to deploy simple Windows VM and Recovery Services vault configured with the *DefaultPolicy* for *Protection*.

```
{  
  "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",  
  "contentVersion": "1.0.0.0",  
  "metadata": {  
    "_generator": {  
      "name": "bicep",  
      "version": "0.5.6.12127",  
      "templateHash": "12431143174203400392"  
    }  
  },  
  "parameters": {  
    " projectName": {  
      "type": "string",  
      "maxLength": 8,  
      "metadata": {  
        "description": "Specifies a name for generating resource names."  
      }  
    },  
    " location": {  
      "type": "string",  
      "defaultValue": "[resourceGroup().location]",  
      "metadata": {  
        "description": "Specifies the location for all resources."  
      }  
    },  
    " adminUsername": {  
      "type": "string",  
      "metadata": {  
        "description": "The administrator's user name for the VM."  
      }  
    },  
    " adminPassword": {  
      "type": "secureString",  
      "metadata": {  
        "description": "The administrator's password for the VM."  
      }  
    }  
  },  
  "resources": [  
    {  
      "type": "Microsoft.RecoveryServices/vaults",  
      "name": "[parameters('projectName')]",  
      "apiVersion": "2019-06-01",  
      "location": "[parameters('location')]",  
      "properties": {  
        "tags": {  
          "Owner": "CloudSkills",  
          "Project": "VM Backup"  
        },  
        "protectionPolicy": {  
          "type": "Default",  
          "retentionPolicy": {  
            "days": 30  
          }  
        }  
      }  
    },  
    {  
      "type": "Microsoft.Compute/virtualMachines",  
      "name": "[parameters('projectName')]-vm",  
      "apiVersion": "2019-06-01",  
      "location": "[parameters('location')]",  
      "dependsOn": "[resourceId('Microsoft.RecoveryServices/vaults', parameters('projectName'))]",  
      "properties": {  
        "osProfile": {  
          "computerName": "[parameters('projectName')]-vm",  
          "adminUsername": "[parameters('adminUsername')]",  
          "adminPassword": "[parameters('adminPassword')]"  
        },  
        "hardwareProfile": {  
          "vmSize": "Standard_DS1_v2"  
        },  
        "storageProfile": {  
          "imageReference": {  
            "uri": "https://azuresamplesv2.blob.core.windows.net/quickstarts/vm-quickstart-image.vhd",  
            "offer": "WindowsServer",  
            "publisher": "MicrosoftWindowsServer",  
            "sku": "WindowsServer-2019-Datacenter"  
          },  
          "osDisk": {  
            "name": "os-disk",  
            "caching": "None",  
            "createOption": "FromImage",  
            "diskSizeGB": 30  
          },  
          "dataDisks": [  
            {  
              "name": "data-disk",  
              "sizeGB": 100, "lun": 1  
            }  
          ]  
        }  
      }  
    }  
  ]  
}
```

```

    "description": "Specifies the administrator username for the Virtual Machine."
    }
  },
  "adminPassword": {
    "type": "secureString",
    "metadata": {
      "description": "Specifies the administrator password for the Virtual Machine."
    }
  },
  "dnsLabelPrefix": {
    "type": "string",
    "metadata": {
      "description": "Specifies the unique DNS Name for the Public IP used to access the Virtual Machine."
    }
  },
  "vmSize": {
    "type": "string",
    "defaultValue": "Standard_A2",
    "metadata": {
      "description": "Virtual machine size."
    }
  },
  "windowsOSVersion": {
    "type": "string",
    "defaultValue": "2016-Datacenter",
    "allowedValues": [
      "2008-R2-SP1",
      "2012-Datacenter",
      "2012-R2-Datacenter",
      "2016-Nano-Server",
      "2016-Datacenter-with-Containers",
      "2016-Datacenter",
      "2019-Datacenter",
      "2019-Datacenter-Core",
      "2019-Datacenter-Core-smalldisk",
      "2019-Datacenter-Core-with-Containers",
      "2019-Datacenter-Core-with-Containers-smalldisk",
      "2019-Datacenter-smalldisk",
      "2019-Datacenter-with-Containers",
      "2019-Datacenter-with-Containers-smalldisk"
    ],
    "metadata": {
      "description": "Specifies the Windows version for the VM. This will pick a fully patched image of this given Windows version."
    }
  },
  "variables": {
    "storageAccountName": "[format('{0}store', parameters('projectName'))]",
    "networkInterfaceName": "[format('{0}-nic', parameters('projectName'))]",
    "vNetAddressPrefix": "10.0.0.0/16",
    "vNetSubnetName": "default",
    "vNetSubnetAddressPrefix": "10.0.0.0/24",
    "publicIPAddressName": "[format('{0}-ip', parameters('projectName'))]",
    "vmName": "[format('{0}-vm', parameters('projectName'))]",
    "vNetName": "[format('{0}-vnet', parameters('projectName'))]",
    "vaultName": "[format('{0}-vault', parameters('projectName'))]",
    "backupFabric": "Azure",
    "backupPolicyName": "DefaultPolicy",
    "protectionContainer": "[format('iaasvmcontainer;iaasvmcontainerv2;{0};{1}', resourceGroup().name, variables('vmName'))]",
    "protectedItem": "[format('vm;iaasvmcontainerv2;{0};{1}', resourceGroup().name, variables('vmName'))]",
    "networkSecurityGroupName": "default-NSG"
  },
  "resources": [
  {
    "type": "Microsoft.Storage/storageAccounts",
    "apiVersion": "2021-08-01",
    "name": "EugenieBlosStorageAccountNameV1"
  }
]
}

```

```

    "name": "[variables('storageAccountName')]",
    "location": "[parameters('location')]",
    "sku": {
        "name": "Standard_LRS"
    },
    "kind": "Storage",
    "properties": {}
},
{
    "type": "Microsoft.Network/publicIPAddresses",
    "apiVersion": "2021-05-01",
    "name": "[variables('publicIPAddressName')]",
    "location": "[parameters('location')]",
    "properties": {
        "publicIPAllocationMethod": "Dynamic",
        "dnsSettings": {
            "domainNameLabel": "[parameters('dnsLabelPrefix')]"
        }
    }
},
{
    "type": "Microsoft.Network/networkSecurityGroups",
    "apiVersion": "2021-05-01",
    "name": "[variables('networkSecurityGroupName')]",
    "location": "[parameters('location')]",
    "properties": {
        "securityRules": [
            {
                "name": "default-allow-3389",
                "properties": {
                    "priority": 1000,
                    "access": "Allow",
                    "direction": "Inbound",
                    "destinationPortRange": "3389",
                    "protocol": "Tcp",
                    "sourceAddressPrefix": "*",
                    "sourcePortRange": "*",
                    "destinationAddressPrefix": "*"
                }
            }
        ]
    }
},
{
    "type": "Microsoft.Network/virtualNetworks",
    "apiVersion": "2021-05-01",
    "name": "[variables('vNetName')]",
    "location": "[parameters('location')]",
    "properties": {
        "addressSpace": {
            "addressPrefixes": [
                "[variables('vNetAddressPrefix')]"
            ]
        },
        "subnets": [
            {
                "name": "[variables('vNetSubnetName')]",
                "properties": {
                    "addressPrefix": "[variables('vNetSubnetAddressPrefix')]",
                    "networkSecurityGroup": {
                        "id": "[resourceId('Microsoft.Network/networkSecurityGroups',
variables('networkSecurityGroupName'))]"
                    }
                }
            }
        ]
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkSecurityGroups', variables('networkSecurityGroupName'))]"
    ]
}

```

```

        ],
    },
    {
        "type": "Microsoft.Network/networkInterfaces",
        "apiVersion": "2021-05-01",
        "name": "[variables('networkInterfaceName')]",
        "location": "[parameters('location')]",
        "properties": {
            "ipConfigurations": [
                {
                    "name": "ipconfig1",
                    "properties": {
                        "privateIPAllocationMethod": "Dynamic",
                        "publicIPAddress": {
                            "id": "[resourceId('Microsoft.Network/publicIPAddresses',
variables('publicIPAddressName'))]"
                        },
                        "subnet": {
                            "id": "[format('{0}/subnets/{1}', resourceId('Microsoft.Network/virtualNetworks',
variables('vNetName')), variables('vNetSubnetName'))]"
                        }
                    }
                }
            ]
        },
        "dependsOn": [
            "[resourceId('Microsoft.Network/publicIPAddresses', variables('publicIPAddressName'))]",
            "[resourceId('Microsoft.Network/virtualNetworks', variables('vNetName'))]"
        ]
    },
    {
        "type": "Microsoft.Compute/virtualMachines",
        "apiVersion": "2021-11-01",
        "name": "[variables('vmName')]",
        "location": "[parameters('location')]",
        "properties": {
            "hardwareProfile": {
                "vmSize": "[parameters('vmSize')]"
            },
            "osProfile": {
                "computerName": "[variables('vmName')]",
                "adminUsername": "[parameters('adminUsername')]",
                "adminPassword": "[parameters('adminPassword')]"
            },
            "storageProfile": {
                "imageReference": {
                    "publisher": "MicrosoftWindowsServer",
                    "offer": "WindowsServer",
                    "sku": "[parameters('windowsOSVersion')]",
                    "version": "latest"
                },
                "osDisk": {
                    "createOption": "FromImage"
                },
                "dataDisks": [
                    {
                        "diskSizeGB": 1023,
                        "lun": 0,
                        "createOption": "Empty"
                    }
                ]
            },
            "networkProfile": {
                "networkInterfaces": [
                    {
                        "id": "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]"
                    }
                ]
            }
        }
    }
]
}

```

```

    "diagnosticsProfile": {
        "bootDiagnostics": {
            "enabled": true,
            "storageUri": "[reference(resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))).primaryEndpoints.blob]"
        }
    },
    "dependsOn": [
        "[resourceId('Microsoft.Network/networkInterfaces', variables('networkInterfaceName'))]",
        "[resourceId('Microsoft.Storage/storageAccounts', variables('storageAccountName'))]"
    ]
},
{
    "type": "Microsoft.RecoveryServices/vaults",
    "apiVersion": "2022-01-01",
    "name": "[variables('vaultName')]",
    "location": "[parameters('location')]",
    "sku": {
        "name": "RS0",
        "tier": "Standard"
    },
    "properties": {}
},
{
    "type": "Microsoft.RecoveryServices/vaults/backupFabrics/protectionContainers/protectedItems",
    "apiVersion": "2022-01-01",
    "name": "[format('{0}/{1}/{2}/{3}', variables('vaultName'), variables('backupFabric'), variables('protectionContainer'), variables('protectedItem'))]",
    "properties": {
        "protectedItemType": "Microsoft.Compute/virtualMachines",
        "policyId": "[format('{0}/backupPolicies/{1}', resourceId('Microsoft.RecoveryServices/vaults', variables('vaultName')), variables('backupPolicyName'))]",
        "sourceResourceId": "[resourceId('Microsoft.Compute/virtualMachines', variables('vmName'))]"
    },
    "dependsOn": [
        "[resourceId('Microsoft.RecoveryServices/vaults', variables('vaultName'))]",
        "[resourceId('Microsoft.Compute/virtualMachines', variables('vmName'))]"
    ]
}
]
}

```

The resources defined in the template are:

- [Microsoft.Storage/storageAccounts](#)
- [Microsoft.Network/publicIPAddresses](#)
- [Microsoft.Network/networkSecurityGroups](#)
- [Microsoft.Network/virtualNetworks](#)
- [Microsoft.Network/networkInterfaces](#)
- [Microsoft.Compute/virtualMachines](#)
- [Microsoft.RecoveryServices/vaults](#)
- [Microsoft.RecoveryServices/vaults/backupFabrics/protectionContainers/protectedItems](#)

## Deploy the template

To deploy the template, select **Try it** to open the Azure Cloud Shell, and then paste the following PowerShell script into the shell window. To paste the code, right-click the shell window and then select **Paste**.

```

$ projectName = Read-Host -Prompt "Enter a project name (limited to eight characters) that is used to generate Azure resource names"
$ location = Read-Host -Prompt "Enter the location (for example, centralus)"
$ adminUsername = Read-Host -Prompt "Enter the administrator username for the virtual machine"
$ adminPassword = Read-Host -Prompt "Enter the administrator password for the virtual machine" -AsSecureString
$ dnsPrefix = Read-Host -Prompt "Enter the unique DNS Name for the Public IP used to access the virtual machine"

$ resourceGroupName = "${projectName}rg"
$ templateUri = "https://raw.githubusercontent.com/Azure/azure-quickstart-templates/master/quickstarts/microsoft.recoveryservices/recovery-services-create-vm-and-configure-backup/azuredeploy.json"

New-AzResourceGroup -Name $resourceGroupName -Location $location
New-AzResourceGroupDeployment -ResourceGroupName $resourceGroupName -TemplateUri $templateUri - projectName $projectName -adminUsername $adminUsername -adminPassword $adminPassword -dnsLabelPrefix $dnsPrefix

```

Azure PowerShell is used to deploy the ARM template in this quickstart. The [Azure portal](#), [Azure CLI](#), and [REST API](#) can also be used to deploy templates.

## Validate the deployment

### Start a backup job

The template creates a VM and enables backup on the VM. After you deploy the template, you need to start a backup job. For more information, see [Start a backup job](#).

### Monitor the backup job

To monitor the backup job, see [Monitor the backup job](#).

## Clean up resources

If you no longer need to back up the VM, you can clean it up.

- If you want to try out restoring the VM, skip the cleanup.
- If you used an existing VM, you can skip the final `Remove-AzResourceGroup` cmdlet to leave the resource group and VM in place.

Disable protection, remove the restore points and vault. Then delete the resource group and associated VM resources, as follows:

```

Disable-AzRecoveryServicesBackupProtection -Item $item -RemoveRecoveryPoints
$vault = Get-AzRecoveryServicesVault -Name "myRecoveryServicesVault"
Remove-AzRecoveryServicesVault -Vault $vault
Remove-AzResourceGroup -Name "myResourceGroup"

```

## Next steps

In this quickstart, you created a Recovery Services vault, enabled protection on a VM, and created the initial recovery point.

- [Learn how](#) to back up VMs in the Azure portal.
- [Learn how](#) to quickly restore a VM
- [Learn how](#) to create ARM templates.

# Use Azure portal to back up multiple virtual machines

9/21/2022 • 7 minutes to read • [Edit Online](#)

When you back up data in Azure, you store that data in an Azure resource called a Recovery Services vault. The Recovery Services vault resource is available from the Settings menu of most Azure services. The benefit of having the Recovery Services vault integrated into the Settings menu of most Azure services is the ease of backing up data. However, working individually with each database or virtual machine in your business is tedious. What if you want to back up the data for all virtual machines in one department, or in one location? It's easy to back up multiple virtual machines by creating a backup policy and applying that policy to the desired virtual machines. This tutorial explains how to:

- Create a Recovery Services vault
- Define a backup policy
- Apply the backup policy to protect multiple virtual machines
- Trigger an on-demand backup job for the protected virtual machines

## Sign in to the Azure portal

Sign in to the [Azure portal](#).

### NOTE

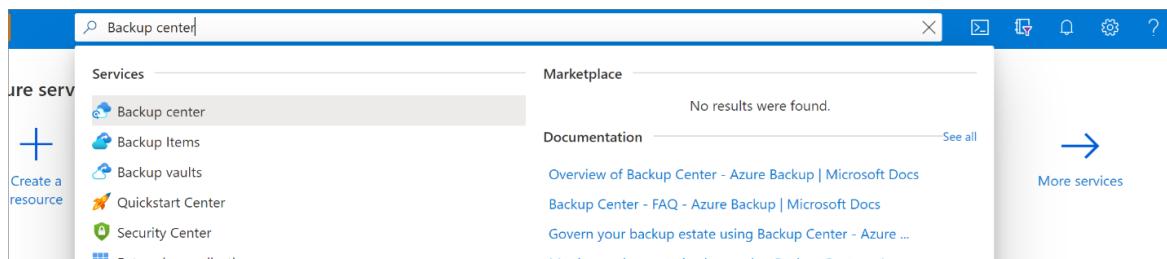
The functionality described in the following sections can also be accessed via [Backup center](#). Backup center is a single unified management experience in Azure. It enables enterprises to govern, monitor, operate, and analyze backups at scale. With this solution, you can perform most of the key backup management operations without being limited to the scope of an individual vault.

## Create a Recovery Services vault

A Recovery Services vault is a management entity that stores recovery points created over time and provides an interface to perform backup-related operations. These operations include taking on-demand backups, performing restores, and creating backup policies.

To create a Recovery Services vault:

1. Sign in to your subscription in the [Azure portal](#).
2. Search for **Backup center** in the Azure portal, and go to the **Backup Center** dashboard.



3. Select **+Vault** from the **Overview** tab.

**Backup center** Microsoft

Search (Ctrl+ /) <> + Backup ⌂ Restore + Policy + Vault ⌂ Refresh

Datasource subscription == **66 selected** Datasource resource group == All Datasource location == All Datasource type == Azure Virtual machines

**Datasource type: Azure Virtual machines**

Overview of Jobs and Backup instances

Jobs (last 24 Hours)				<a href="#">View all</a>
Operation	Failed	In progress	Completed	
Scheduled backup	64	8	246	
On-demand backup	0	0	12	
Restore	1	0	57	

Backup instances			
Azure Virtual machines			
<b>359</b>	Protection configured	<b>338</b>	
	Protection stopped	<b>16</b>	
	Soft deleted	<b>5</b>	
<b>86</b> out of 359	Backup instances with the underlying datasource not found		

#### 4. Select Recovery Services vault > Continue.

Home > Backup center > Start: Create Vault ...

A vault is an entity that stores the backups and restore points created over time. The vault also contains the backup policies that are associated with the protected virtual machines. Proceed to vault creation by selecting vault type.

Vault Type

Recovery Services vault

Supported datasources

- ✓ Azure Virtual machines
- ✓ SQL in Azure VM
- ✓ Azure Files (Azure Storage)
- ✓ SAP HANA in Azure VM
- ✓ Azure Backup Server
- ✓ Azure Backup Agent
- ✓ DPM

Backup vault

Supported datasources

- ✓ Azure Database for PostgreSQL servers (Preview)
- ✓ Azure Blobs (Azure Storage)
- ✓ Azure Disks

Learn more about Backup vault. [Click here](#).

Learn more about Recovery Services vault. [Click here](#).

[Continue](#) [Cancel](#)

#### 5. The Recovery Services vault dialog opens. Provide the following values:

- Subscription:** Choose the subscription to use. If you're a member of only one subscription, you'll see that name. If you're not sure which subscription to use, use the default (suggested) subscription. There are multiple choices only if your work or school account is associated with more than one Azure subscription.
- Resource group:** Use an existing resource group or create a new one. To see the list of available resource groups in your subscription, select **Use existing**, and then select a resource from the dropdown list. To create a new resource group, select **Create new** and enter the name. For more information about resource groups, see [Azure Resource Manager overview](#).
- Vault name:** Enter a friendly name to identify the vault. The name must be unique to the Azure

subscription. Specify a name that has at least 2 but not more than 50 characters. The name must start with a letter and consist only of letters, numbers, and hyphens.

- **Region:** Select the geographic region for the vault. For you to create a vault to help protect any data source, the vault *must* be in the same region as the data source.

#### IMPORTANT

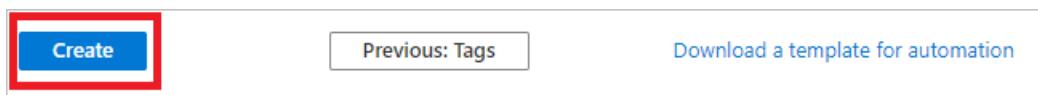
If you're not sure of the location of your data source, close the dialog. Go to the list of your resources in the portal. If you have data sources in multiple regions, create a Recovery Services vault for each region. Create the vault in the first location before you create the vault for another location. There's no need to specify storage accounts to store the backup data. The Recovery Services vault and Azure Backup handle that automatically.

The screenshot shows the 'Create Recovery Services vault' wizard in the 'Basics' step. It includes fields for 'Subscription' (dropdown), 'Resource group' (dropdown with 'Create new' option), 'Vault name' (text input placeholder 'Enter the name for your vault.'), and 'Region' (dropdown set to 'East US'). Navigation buttons at the bottom are 'Review + create' (highlighted in red) and 'Next: Tags'.

6. After you provide the values, select **Review + create**.



7. When you're ready to create the Recovery Services vault, select **Create**.



8. It can take a while to create the Recovery Services vault. Monitor the status notifications in the **Notifications** area at the upper-right corner of the portal. After your vault is created, it's visible in the list of Recovery Services vaults. If you don't see your vault, select **Refresh**.

The screenshot shows the 'Recovery Services vaults' list page. At the top right, there is a 'Refresh' button highlighted with a red box. Other buttons include '+ Add', 'Edit columns', 'Try preview', and 'Assign tags'. The Microsoft logo is also visible.

When you create a Recovery Services vault, by default the vault has geo-redundant storage. To provide data resiliency, geo-redundant storage replicates the data multiple times across two Azure regions.

## Set backup policy to protect VMs

After creating the Recovery Services vault, the next step is to configure the vault for the type of data, and to set the backup policy. Backup policy is the schedule for how often and when recovery points are taken. Policy also includes the retention range for the recovery points. For this tutorial, let's assume your business is a sports complex with a hotel, stadium, and restaurants and concessions, and you're protecting the data on the virtual machines. The following steps create a backup policy for the financial data.

To set a backup policy to your Azure VMs, follow these steps:

1. Go to **Backup center** and click **+ Backup** from the **Overview** tab.

The screenshot shows the Azure Backup center interface. The left sidebar contains navigation links like Overview, Getting started, Community, Manage (Backup instances, Backup policies, Vaults), Monitoring + reporting (Backup jobs, Backup reports), Policy and compliance (Backup compliance, Azure policies for backup, Protectable datasources), and Support + troubleshooting (New support request). The main area is titled 'Datasource type: Azure Virtual machines' and shows 'Overview of Jobs and Backup instances'. It includes a table for 'Jobs (last 24 Hours)' with columns for Operation (Scheduled backup, On-demand backup, Restore), Failed, In progress, and Completed counts. Below this is a 'Backup instances' section for 'Azure Virtual machines' with counts for Protection configured (359), Protection stopped (16), Soft deleted (5), and a note about 86 instances not found. A red box highlights the '+ Backup' button in the top navigation bar.

2. Select **Azure Virtual machines** as the **Datasource type** and select the vault you have created. Then click **Continue**.

The screenshot shows the 'Initiate: Configure Backup' step. It has three dropdown fields: 'Datasource type' set to 'Azure Virtual machines', 'Vault type' set to 'Recovery Services Vault', and 'Vault \*' with a placeholder 'Select a Vault' and a 'Select' button. A blue info box at the bottom states: 'Selected vault is maintained in **Recovery Services Vault**, you will be redirected to Recovery Services Vault for configuring backup. [Learn more](#)'.

3. Assign a Backup policy.

- The default policy backs up the VM once a day. The daily backups are retained for 30 days. Instant recovery snapshots are retained for two days.

Home > Backup center > Start: Configure Backup >

## Configure Backup

VaultWithTag

Backup policy \* ⓘ DefaultPolicy  Create a new policy

**Policy Details**

Full Backup      **Backup Frequency**  
Daily at 6:00 PM UTC

Instant Restore  
Retain instant recovery snapshot(s) for 2 day(s)

Retention of daily backup point  
Retain backup taken every day at 6:00 PM for 30 Day(s)

**Virtual machines**

Name	Resource Group	OS Disk Only
No Virtual machines selected.		

Add

**OS Disk only backup** option allows you to backup Azure Virtual Machine with only OS disk and exclude all the data disks. You can use Selective Disk Backup feature through Powershell or CLI to include or exclude specific data disks. Know more about Selective Disk Backup feature, its limitation and pricing- [Learn more](#).

- If you don't want to use the default policy, select **Create New**, and create a custom policy as described in the next procedure.

4. Under **Virtual Machines**, select Add.

### Virtual Machines

Virtual machine name	Resource Group
No Virtual Machines Selected	

Add

5. The **Select virtual machines** pane will open. Select the VMs you want to back up using the policy. Then select **OK**.

- The selected VMs are validated.
- You can only select VMs in the same region as the vault.
- VMs can only be backed up in a single vault.

## Select virtual machines

X

Virtual machine name	Resource Group
myVM	myResourceGroup
myVMH1	myResourceGroup
myVMR1	myResourceGroup

OK

### NOTE

All the VMs in the same region and subscription as that of the vault are available to configure backup. When configuring backup, you can browse to the virtual machine name and its resource group, even though you don't have the required permission on those VMs. If your VM is in soft deleted state, then it won't be visible in this list. If you need to re-protect the VM, then you need to wait for the soft delete period to expire or undelete the VM from the soft deleted list. For more information, see [the soft delete for VMs article](#).

6. In **Backup**, select **Enable backup**. This deploys the policy to the vault and to the VMs, and installs the backup extension on the VM agent running on the Azure VM.

After enabling backup:

- The Backup service installs the backup extension whether or not the VM is running.
- An initial backup will run in accordance with your backup schedule.
- When backups run, note that:
  - A VM that's running has the greatest chance for capturing an application-consistent recovery point.
  - However, even if the VM is turned off, it's backed up. Such a VM is known as an offline VM. In this case, the recovery point will be crash-consistent.
- Explicit outbound connectivity isn't required to allow backup of Azure VMs.

## Initial backup

You've enabled backup for the Recovery Services vaults, but an initial backup hasn't been created. It's a disaster recovery best practice to trigger the first backup, so that your data is protected.

The initial backup will run in accordance with the schedule, but you can run it immediately as follows:

1. Go to **Backup center** and select the **Backup Instances** menu item.
2. Select **Azure Virtual machines** as the **Datasource type**. Then search for the VM that you have configured for backup.
3. Right-click the relevant row or select the more icon (...), and then click **Backup Now**.
4. In **Backup Now**, use the calendar control to select the last day that the recovery point should be retained. Then select **OK**.

5. Monitor the portal notifications. To monitor the job progress, go to **Backup center > Backup Jobs** and filter the list for **In progress** jobs. Depending on the size of your VM, creating the initial backup may take a while.

## Clean up resources

If you plan to continue on to work with subsequent tutorials, don't clean up the resources created in this tutorial. If you don't plan to continue, use the following steps to delete all resources created by this tutorial in the Azure portal.

1. On the **myRecoveryServicesVault** dashboard, select **3** under **Backup Items** to open the **Backup Items** menu.

The screenshot shows the Microsoft Azure portal interface for the 'myRecoveryServicesVault' Recovery Services vault. The left sidebar contains navigation links for 'All resources', 'Overview', 'Activity log', 'Access control (IAM)', 'Tags', 'Diagnose and solve problems', 'Settings' (Identity, Private endpoint connections, Properties, Locks, Export template), 'Getting started' (Backup, Site Recovery), 'Protected items' (Backup items, Replicated items), 'Manage' (Backup policies, Backup Infrastructure, Site Recovery infrastructure), and 'Usage'. The main content area displays the vault's details: Resource group (myResourceGroup), Location (East US), Subscription (subscription-name, subscription-id). It includes tabs for 'Overview', 'Backup' (selected), and 'Site Recovery'. The 'Monitoring' section shows 'Backup Alerts (last 24 hours)' for Critical (0) and Warning (0) levels. The 'Usage' section shows 'Backup items' (3 highlighted with a red box) and 'Backup Storage' (Cloud - LRS: 0 B, Cloud - GRS: 0 B). A 'Backup Pre-Check Status (Azure VMs)' chart indicates 0 Critical and 0 Warning errors.

2. On the **Backup Items** menu, select **Azure Virtual Machine** to open the list of virtual machines associated with the vault.

The screenshot shows the 'myRecoveryServicesVault | Backup items' page. The left sidebar is identical to the previous dashboard. The main content area lists 'BACKUP MANAGEMENT TYPE' categories: 'Azure Virtual Machine' (selected and highlighted with a red box), 'SAP HANA in Azure VM', 'SQL in Azure VM', 'Azure Storage (Azure Files)', 'DPM', 'Azure Backup Server', and 'Azure Backup Agent'. To the right, a 'BACKUP ITEM COUNT' column shows values: 3 for Azure Virtual Machine, 0 for SAP HANA in Azure VM, 0 for SQL in Azure VM, 0 for Azure Storage (Azure Files), 0 for DPM, 0 for Azure Backup Server, and 0 for Azure Backup Agent.

The **Backup Items** list opens.

3. In the **Backup Items** menu, select the ellipsis to open the Context menu.

All services > myRecoveryServicesVault | Backup items >

## Backup Items (Azure Virtual Machine)

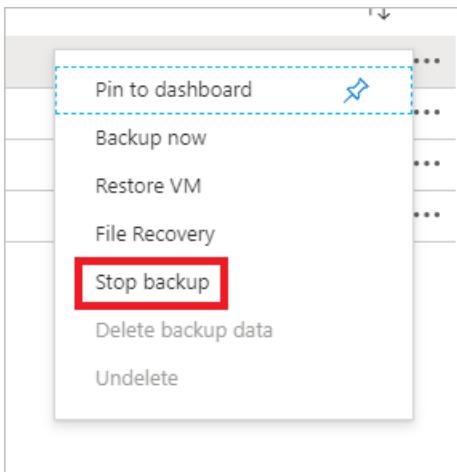
myRecoveryServicesVault

Refresh Add Filter

Fetching data from service completed.

Name	Resource Group	Backup Pre-Check	Last Backup Status	Latest restore point	...
myVM	myResourceGroup	Passed	Success	7/26/2020, 4:23:25 PM	

4. On the context menu, select **Stop backup** to open Stop Backup menu.



5. In the **Stop Backup** menu, select the upper drop-down menu and choose **Delete Backup Data**.
6. In the **Type the name of the Backup item** dialog, type *myVM*.
7. Once the backup item is verified (a check mark appears), **Stop backup** button is enabled. Select **Stop Backup** to stop the policy and delete the restore points.

### Stop Backup

myVM

Delete Backup Data

This option will stop all scheduled backup jobs and delete backup data. Learn more <https://aka.ms/SoftDeleteCloudWorkloads>

Type the name of Backup Item \*

Reason

Comments

Stop backup

## NOTE

Deleted items are retained in the soft delete state for 14 days. Only after that period can the vault be deleted. For more information, see [Delete an Azure Backup Recovery Services vault](#).

- When there are no more items in the vault, select **Delete**.

The screenshot shows the Azure Recovery Services vault overview page. On the left, a navigation menu includes options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Properties, Locks, Automation script, Backup, Site Recovery, Backup Jobs, and Site Recovery Jobs. The main area displays monitoring and usage statistics. Under Monitoring, it shows 'Backup Alerts (last 24 hours)' with 0 Critical and 0 Warning alerts. Under Usage, it shows 'Backup items' (0) and 'Backup Storage' (Cloud - LRS: 0 B, Cloud - GRS: 0 B). A large circular gauge chart indicates 'Backup Pre-Check Status (Azure VMs)' with 0 Critical and 0 Warning issues.

Once the vault is deleted, you'll return to the list of Recovery Services vaults.

## Next steps

In this tutorial, you used the Azure portal to:

- Create a Recovery Services vault
- Set the vault to protect virtual machines
- Create a custom backup and retention policy
- Assign the policy to protect multiple virtual machines
- Trigger an on-demand back up for virtual machines

Continue to the next tutorial to restore an Azure virtual machine from disk.

[Restore VMs using CLI](#)

# How to restore Azure VM data in Azure portal

9/21/2022 • 21 minutes to read • [Edit Online](#)

This article describes how to restore Azure VM data from the recovery points stored in [Azure Backup Recovery Services vaults](#).

## Restore options

Azure Backup provides several ways to restore a VM.

RESTORE OPTION	DETAILS
<b>Create a new VM</b>	<p>Quickly creates and gets a basic VM up and running from a restore point.</p> <p>You can specify a name for the VM, select the resource group and virtual network (VNet) in which it will be placed, and specify a storage account for the restored VM. The new VM must be created in the same region as the source VM.</p> <p>If a VM restore fails because an Azure VM SKU wasn't available in the specified region of Azure, or because of any other issues, Azure Backup still restores the disks in the specified resource group.</p>
<b>Restore disk</b>	<p>Restores a VM disk, which can then be used to create a new VM.</p> <p>Azure Backup provides a template to help you customize and create a VM.</p> <p>The restore job generates a template that you can download and use to specify custom VM settings, and create a VM.</p> <p>The disks are copied to the Resource Group you specify.</p> <p>Alternatively, you can attach the disk to an existing VM, or create a new VM using PowerShell.</p> <p>This option is useful if you want to customize the VM, add configuration settings that weren't there at the time of backup, or add settings that must be configured using the template or PowerShell.</p>

RESTORE OPTION	DETAILS
<b>Replace existing</b>	<p>You can restore a disk, and use it to replace a disk on the existing VM.</p> <p>The current VM must exist. If it's been deleted, this option can't be used.</p> <p>Azure Backup takes a snapshot of the existing VM before replacing the disk, and stores it in the staging location you specify. Existing disks connected to the VM are replaced with the selected restore point.</p> <p>The snapshot is copied to the vault, and retained in accordance with the retention policy.</p> <p>After the replace disk operation, the original disk is retained in the resource group. You can choose to manually delete the original disks if they aren't needed.</p> <p>Replace existing is supported for unencrypted managed VMs, including VMs <a href="#">created using custom images</a>. It's unsupported for classic VMs, unmanaged VMs, and <a href="#">generalized VMs</a>.</p> <p>If the restore point has more or less disks than the current VM, then the number of disks in the restore point will only reflect the VM configuration.</p> <p>Replace existing is also supported for VMs with linked resources, like <a href="#">user-assigned managed-identity</a> or <a href="#">Key Vault</a>.</p>
<b>Cross Region (secondary region)</b>	<p>Cross Region restore can be used to restore Azure VMs in the secondary region, which is an <a href="#">Azure paired region</a>.</p> <p>You can restore all the Azure VMs for the selected recovery point if the backup is done in the secondary region.</p> <p>During the backup, snapshots aren't replicated to the secondary region. Only the data stored in the vault is replicated. So secondary region restores are only <a href="#">vault tier</a> restores. The restore time for the secondary region will be almost the same as the vault tier restore time for the primary region.</p> <p>This feature is available for the options below:</p> <ul style="list-style-type: none"> <li>- <a href="#">Create a VM</a></li> <li>- <a href="#">Restore Disks</a></li> </ul> <p>We don't currently support the <a href="#">Replace existing disks</a> option.</p> <p><b>Permissions</b> The restore operation on secondary region can be performed by Backup Admins and App admins.</p>

RESTORE OPTION	DETAILS
Cross Subscription Restore	<p>Allows you to restore Azure Virtual Machines or disks to any subscription (as per the Azure RBAC capabilities) from restore points.</p> <p>You can trigger Cross Subscription Restore for managed virtual machines only.</p> <p>Cross Subscription Restore is supported for <a href="#">Restore with Managed System Identities (MSI)</a>.</p> <p>It's unsupported from <a href="#">snapshots</a> and <a href="#">secondary region</a> restores.</p> <p>It's unsupported for <a href="#">Encrypted Azure VMs</a> and <a href="#">Trusted Launch VMs</a>.</p>

#### TIP

To receive alerts/notifications when a restore operation fails, use [Azure Monitor alerts for Azure Backup](#). This helps you to monitor such failures and take necessary actions to remediate the issues.

#### NOTE

You can also recover specific files and folders on an Azure VM. [Learn more](#).

## Storage accounts

Some details about storage accounts:

- **Create VM:** When you create a new VM, the VM will be placed in the storage account you specify.
- **Restore disk:** When you restore a disk, the disk is copied to the storage account you specify. The restore job generates a template that you can download and use to specify custom VM settings. This template is placed in the specified storage account.
- **Replace disk:** When you replace a disk in an existing VM, Azure Backup takes a snapshot of the existing VM before replacing the disk. The snapshot is also copied to the Recovery Services vault through data transfer, as a background process. However, once the snapshot phase is completed, the replace disks operation is triggered. After the replace disk operation, the disks of the source Azure VM are left in the specified Resource group for your operation and the VHDs are stored in the specified storage account. You can choose to delete or retain these VHDs and disks.
- **Storage account location:** The storage account must be in the same region as the vault. Only these accounts are displayed. If there are no storage accounts in the location, you need to create one.
- **Storage type:** Blob storage isn't supported.
- **Storage redundancy:** Zone redundant storage (ZRS) isn't supported. The replication and redundancy information for the account is shown in parentheses after the account name.
- **Premium storage:**
  - When you restore non-premium VMs, premium storage accounts aren't supported.
  - When you restore managed VMs, premium storage accounts configured with network rules aren't supported.

## Before you start

To restore a VM (create a new VM), make sure you have the correct Azure role-based access control (Azure RBAC) permissions for the Restore VM operation.

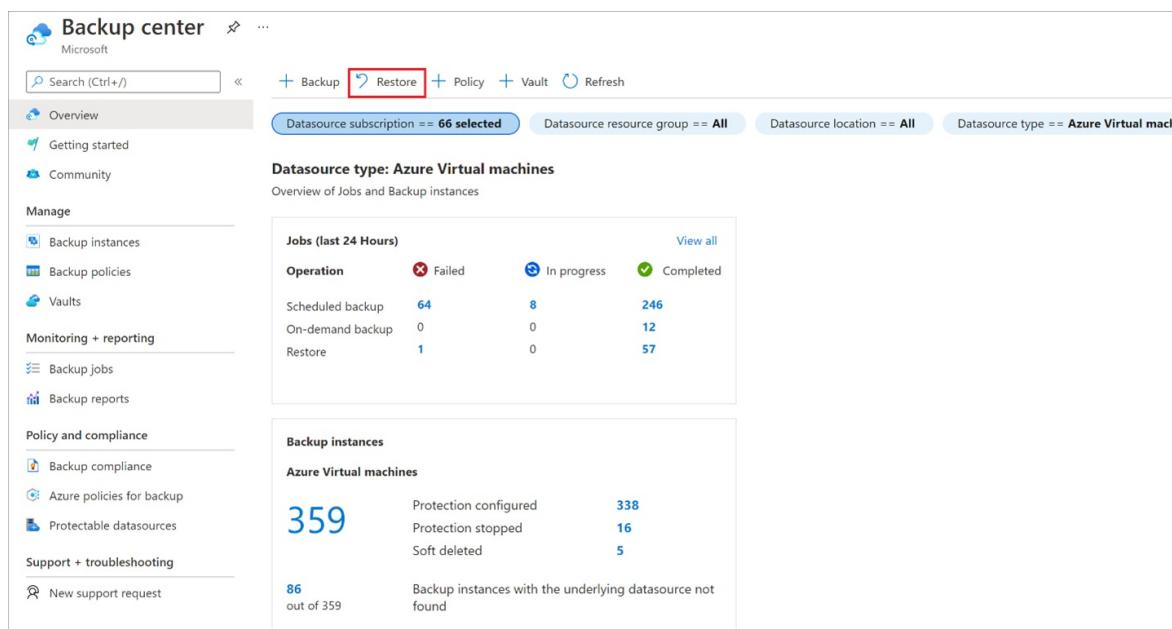
If you don't have permissions, you can [restore a disk](#), and then after the disk is restored, you can [use the template](#) that was generated as part of the restore operation to create a new VM.

#### NOTE

The functionality described in the following sections can also be accessed via [Backup center](#). Backup center is a single unified management experience in Azure. It enables enterprises to govern, monitor, operate, and analyze backups at scale. With this solution, you can perform most of the key backup management operations without being limited to the scope of an individual vault.

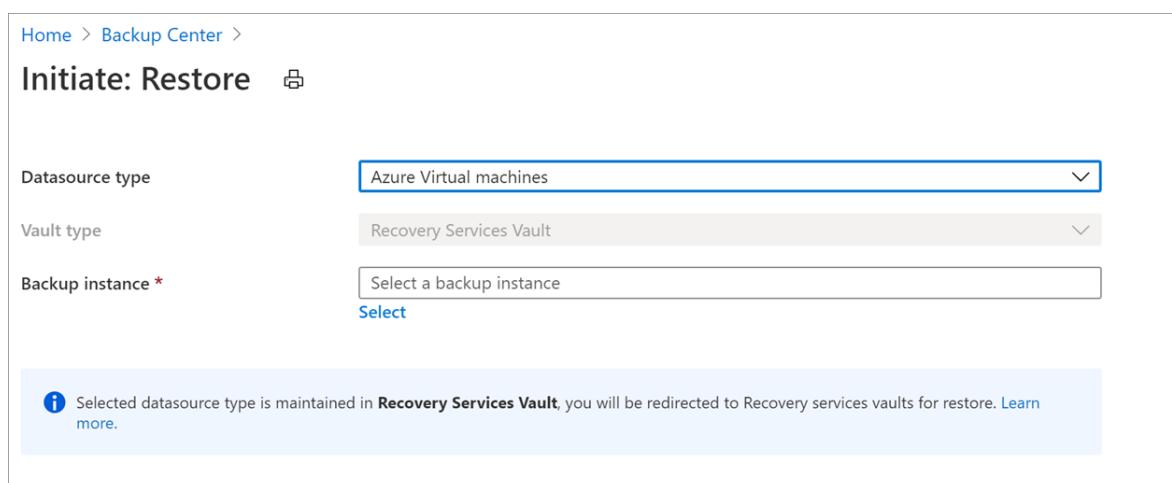
## Select a restore point

1. Navigate to **Backup center** in the Azure portal and click **Restore** from the **Overview** tab.



The screenshot shows the Azure Backup center interface. The left sidebar includes links for Overview, Getting started, Community, Manage (Backup instances, Backup policies, Vaults, Monitoring + reporting, Backup jobs, Backup reports), Policy and compliance (Backup compliance, Azure policies for backup, Protectable datasources), Support + troubleshooting, and New support request. The main area displays 'Jobs (last 24 Hours)' with a table showing Failed (64), In progress (8), and Completed (246) scheduled backups, and one restore job. Below this is a 'Backup instances' section for Azure Virtual machines, showing 359 total instances with 338 protected, 16 stopped, and 5 soft deleted. A note indicates 86 instances are not found. The top navigation bar has tabs for Backup, Restore (highlighted with a red box), Policy, Vault, and Refresh, along with filters for Datasource subscription (66 selected), Datasource resource group (All), Datasource location (All), and Datasource type (Azure Virtual machines).

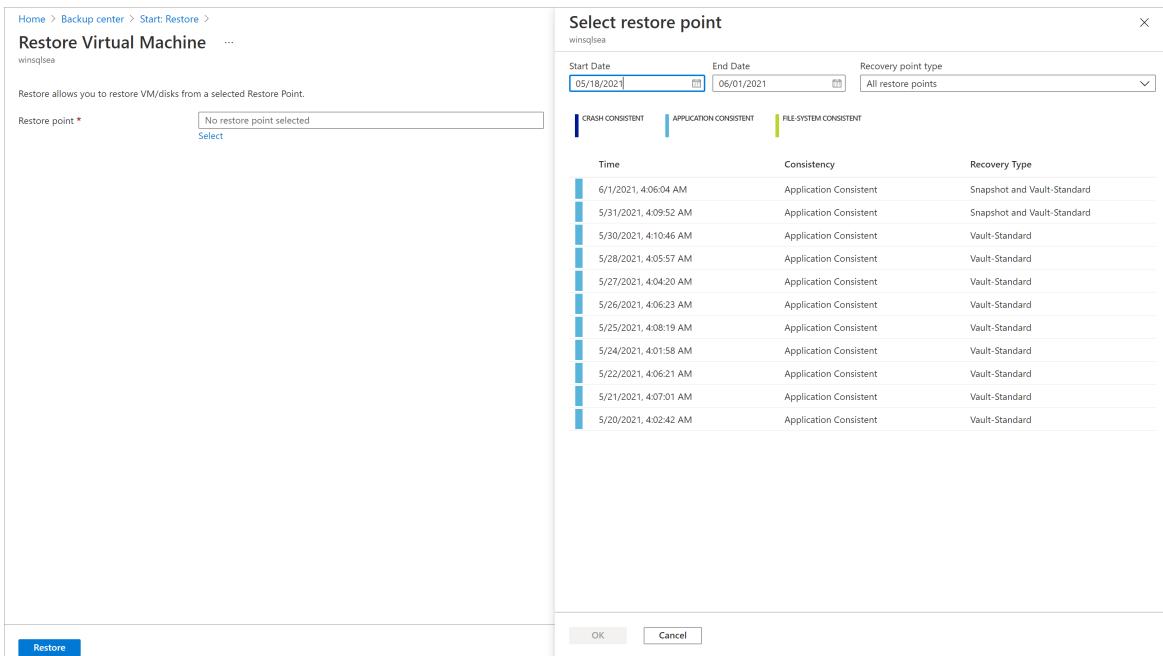
2. Select **Azure Virtual machines** as the **Datasource type**, and then select a **Backup instance**.



The screenshot shows the 'Initiate: Restore' step. It has fields for 'Datasource type' (set to 'Azure Virtual machines'), 'Vault type' (set to 'Recovery Services Vault'), and 'Backup instance \*' (with a dropdown menu open). A note at the bottom says: 'Selected datasource type is maintained in Recovery Services Vault, you will be redirected to Recovery services vaults for restore. [Learn more.](#)'

3. Select a VM and click **Continue**.

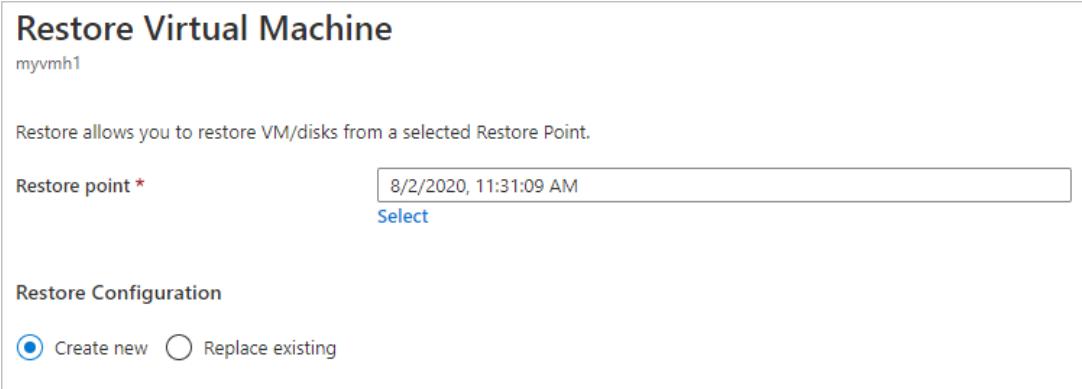
4. In the next screen that appears, select a restore point to use for the recovery.



## Choose a VM restore configuration

1. In **Restore Virtual Machine**, select a restore option:

- **Create new:** Use this option if you want to create a new VM. You can create a VM with simple settings, or restore a disk and create a customized VM.
- **Replace existing:** Use this option if you want to replace disks on an existing VM.



2. Specify settings for your selected restore option.

## Create a VM

As one of the [restore options](#), you can create a VM quickly with basic settings from a restore point.

1. In **Restore Virtual Machine > Create new > Restore Type**, select **Create new virtual machine**.
2. In **Virtual machine name**, specify a VM that doesn't exist in the subscription.
3. In **Resource group**, select an existing resource group for the new VM, or create a new one with a globally unique name. If you assign a name that already exists, Azure assigns the group the same name as the VM.
4. In **Virtual network**, select the VNet in which the VM will be placed. All VNets associated with the subscription in the same location as the vault, which is active and not attached with any affinity group, are displayed. Select the subnet.

The first subnet is selected by default.

5. In **Staging Location**, specify the storage account for the VM. [Learn more](#).

The screenshot shows the 'Restore Virtual Machine' page in the Azure portal. The URL is: Home > myRecoveryServicesVault | Backup items > Backup Items (Azure Virtual Machine) > myVMH1 > Restore Virtual Machine.

Section: myvmh1

Description: Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*: 8/2/2020, 11:31:09 AM (Select)

Restore Configuration:

Create new  Replace existing

**Info:** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type \*: Create new virtual machine

Virtual machine name \*: vm1Restore

Resource group \*: myResourceGroup

Virtual network \*: Test-Resource-Group-vnet (myResourceGroup)

Subnet \*: default

Staging Location \*: testresourcegroupdiag856 (StandardLRS)

[Can't find your storage account ?](#)

**Restore**

6. Choose the required subscription from the **Subscription** drop-down list to restore an Azure VM to a different subscription.

Azure Backup now supports Cross Subscription Restore (CSR), you can now restore an Azure VM using a recovery point from default subscription to another. Default subscription is the subscription where recovery point is available.

The following screenshot lists all subscriptions under the tenant where you've permissions, which enable you to restore the Azure VM to another subscription.

# Restore Virtual Machine

demorestoredvm1

Restore allows you to restore VM/disks from a backup.

**Restore point \***

- Backup\_Canary\_PM\_PPE\_Demo-1
- Backup\_PPE\_Hybrid\_TIP-1
- Backup\_PPE\_SAPHana\_user1-1
- Bing MM Measurement
- CloudAnalytics\_Prod1
- CLOUDBUILD-ANYBUILD-POC-01
- Code generate Test and Infra
- Contoso Infra1
- Core-ES-BranchManagement
- Core-ES-WorkManagement
- Cosmos\_C&E\_Azure\_AzureEngineeringSystems\_100200
- Cosmos\_WDG\_Core\_BnB\_100292
- CRM-DEVTEST-Efun-IDC

**Restore Configuration**

Create new

Replace existing

**To create an alternate configuration when restoring:**

**Restore Type \*** ⓘ

Virtual machine name \* ⓘ

Subscription \* ⓘ

Resource group \* ⓘ

Virtual network \* ⓘ

Subnet \* ⓘ

Staging Location \* ⓘ

**Can't find your storage account ?**

**Identity** The identities listed here are based on the MSI configurations in the corresponding Recovery services vault. [Learn more.](#)

Enabled

**Restore**

## Restore disks

As one of the [restore options](#), you can create a disk from a restore point. Then with the disk, you can do one of the following actions:

- Use the template that's generated during the restore operation to customize settings, and trigger VM deployment. You edit the default template settings, and submit the template for VM deployment.
  - [Attach restored disks](#) to an existing VM.
  - [Create a new VM](#) from the restored disks using PowerShell.

1. In **Restore configuration > Create new > Restore Type**, select **Restore disks**.
  2. In **Resource group**, select an existing resource group for the restored disks, or create a new one with a globally unique name.

3. In **Staging location**, specify the storage account to which to copy the VHDs. [Learn more.](#)

[Home](#) > [Backup center](#) > [Start: Restore](#) >

## Restore Virtual Machine

winsqleus2

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*

11/20/2021, 5:10:18 AM

Select

Data Store

Vault-Standard

### Restore Configuration

Create new

Replace existing

**i** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type \* ⓘ

Restore disks

Resource group \* ⓘ

Select an option

Staging Location \* ⓘ

Select an option

[Can't find your storage account ?](#)

**i** The identities listed here are based on the MSI configurations in the corresponding Recovery services vault. [Learn more.](#)

Identities ⓘ

Disabled

**Restore**

4. Choose the required subscription from the **Subscription** drop-down list to restore the VM disks to a different subscription.

Azure Backup now supports Cross Subscription Restore (CSR). Like Azure VM, you can now restore Azure VM disks using a recovery point from default subscription to another. Default subscription is the subscription where recovery point is available.

5. Select **Restore** to trigger the restore operation.

When your virtual machine uses managed disks and you select the **Create virtual machine** option, Azure Backup doesn't use the specified storage account. In the case of **Restore disks** and **Instant Restore**, the storage account is used only for storing the template. Managed disks are created in the specified resource group. When your virtual machine uses unmanaged disks, they're restored as blobs to the storage account.

While you restore disks for a Managed VM from a Vault-Standard recovery point, it restores the Managed disk and Azure Resource Manager (ARM) templates, along with the VHD files of the disks in staging location. If you restore disks from an Instant recovery point, it restores the Managed disks and ARM templates only.

#### NOTE

- For restoring disk from a Vault-Standard recovery point that is/was greater than 4 TB, Azure Backup doesn't restore the VHD files.
- For information on managed/premium disk performance after restored via Azure Backup, see the [Latency](#) section.

## Use templates to customize a restored VM

After the disk is restored, use the template that was generated as part of the restore operation to customize and create a new VM:

1. In **Backup Jobs**, select the relevant restore job.
2. In **Restore**, select **Deploy Template** to initiate template deployment.

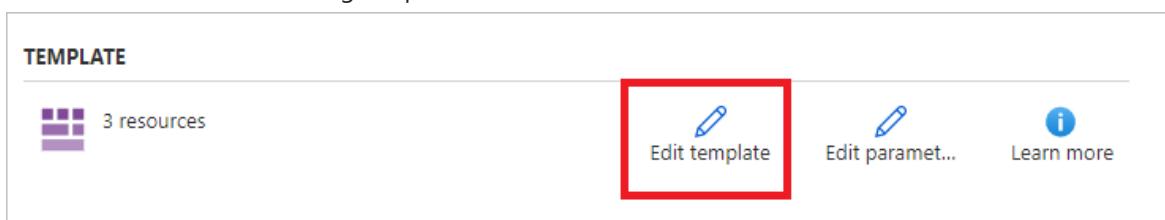


The screenshot shows the 'Restore' blade for a job named 'myvmh1'. The 'Deploy Template' button is highlighted with a red box. Below it, the 'Job Details' table lists various parameters such as Job Type (Recover disks), Target Storage Account Name (testresourcegroupdiag856), Recovery point time (8/2/2020 8:31:09 AM), and Config Blob Uri (https://testresourcegroupdiag856.blob.core.windows.net/myvhm1-ec54d603d8154087a6112a4d26273024/config-myvhm1-044cc161-656d-4618-b741-38fb712d2ec0.json).

#### NOTE

For a shared access signature (SAS) that has **Allow storage account key access** set to disabled, the template won't deploy when you select **Deploy Template**.

3. To customize the VM setting provided in the template, select **Edit template**. If you want to add more customizations, select **Edit parameters**.
  - [Learn more](#) about deploying resources from a custom template.
  - [Learn more](#) about authoring templates.



The screenshot shows the 'TEMPLATE' blade with a '3 resources' count and three buttons: 'Edit template' (highlighted with a red box), 'Edit paramet...', and 'Learn more'.

4. Enter the custom values for the VM, accept the **Terms and Conditions** and select **Purchase**.

## Custom deployment

Deploy from a custom template

### TEMPLATE



3 resources

[Edit template](#)
[Edit param...](#)
[Learn more](#)

### BASICS

Subscription \*

Resource group \*

[Create new](#)

Location

### SETTINGS

Virtual Machine Name \* ⓘ

Virtual Network ⓘ

Virtual Network Resource Group ⓘ

Subnet ⓘ

Os Disk Name ⓘ

Network Interface Prefix Name ⓘ

Public Ip Address Name ⓘ

### TERMS AND CONDITIONS

[Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking "Purchase," I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

 I agree to the terms and conditions stated above

[Purchase](#)

## Replace existing disks

As one of the [restore options](#), you can replace an existing VM disk with the selected restore point. [Review](#) all restore options.

1. In **Restore configuration**, select Replace existing.
2. In **Restore Type**, select Replace disk/s. This is the restore point that will be used replace existing VM disks.
3. In **Staging Location**, specify where snapshots of the current managed disks should be saved during the restore process. [Learn more](#).

Home > Backup center > Start: Restore >

## Restore Virtual Machine

winsqlsea

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*  [Select](#)

Data Store Snapshot and Vault-Standard

To get the list of disks backed up in this recovery point, [click here](#)

Restore Configuration

Create new  Replace existing

**Info** The disk(s) from the selected restore point will replace the disk(s) in your existing VM. Learn more about In-Place Restore.

Restore Type  Replace Disk(s)

Staging Location \*  [Can't find your storage account ?](#)

## Cross Region Restore

As one of the [restore options](#), Cross Region Restore (CRR) allows you to restore Azure VMs in a secondary region, which is an Azure paired region.

To begin using the feature, read the [Before You Begin](#) section.

To see if CRR is enabled, follow the instructions in [Configure Cross Region Restore](#).

### View backup items in secondary region

If CRR is enabled, you can view the backup items in the secondary region.

1. From the portal, go to **Recovery Services vault > Backup items**.
2. Select **Secondary Region** to view the items in the secondary region.

#### NOTE

Only Backup Management Types supporting the CRR feature will be shown in the list. Currently, only support for restoring secondary region data to a secondary region is allowed.

CRR for Azure VMs is supported for Azure Managed VMs (including encrypted Azure VMs). See the [management types that support Cross Region Restore](#).

**CRRIgniteDemoVault - Backup items**  
Recovery Services vault

Search (Ctrl+ /) <> Refresh

Primary Region Secondary Region

BACKUP MANAGEMENT TYPE	BACKUP ITEM COUNT
Azure Virtual Machine	2
SAP HANA in Azure VM	0
SQL in Azure VM	0
DPM	0
Azure Backup Server	0
Azure Backup Agent	0

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

**Settings**

- Properties
- Locks
- Export template

**Getting started**

- Backup
- Site Recovery

**CRRIgniteDemoVault - Backup items**  
Recovery Services vault

Search (Ctrl+ /) <> Refresh

Primary Region Secondary Region

Showing data from "eastus2euap" (Secondary Region)

BACKUP MANAGEMENT TYPE	BACKUP ITEM COUNT
Azure Virtual Machine	2

Overview Activity log Access control (IAM) Tags Diagnose and solve problems

**Settings**

- Properties

## Restore in secondary region

The secondary region restore user experience will be similar to the primary region restore user experience. When configuring details in the Restore Configuration pane to configure your restore, you'll be prompted to provide only secondary region parameters.

Currently, secondary region **RPO** is *36 hours*. This is because the RPO in the primary region is *24 hours* and can take up to *12 hours* to replicate the backup data from the primary to the secondary region.

Home > Backup center >

**Start: Restore** ...

Datasource type	Azure Virtual machines
Vault type	Recovery Services vault
Backup instance *	advm001 Select
Vault	contosotestvault

**Info** The selected vault is enabled with CRR (Cross Region Restore) which enables to restore in secondary region. [Learn more](#).

Restore Region

Primary Region  
 Secondary Region

## Restore Virtual Machine

advm001

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point *	<input type="text" value="5/9/2021, 4:41:30 AM"/> <a href="#">Select</a>												
Data Store	Snapshot and Vault-Standard												
<b>Restore Configuration</b>													
<input checked="" type="radio"/> Create new <input type="radio"/> Replace existing													
<p><b>To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.</b></p> <table border="0"> <tr> <td>Restore Type *</td> <td><input type="text" value="Create new virtual machine"/></td> </tr> <tr> <td>Virtual machine name *</td> <td><input type="text" value="Enter a name"/></td> </tr> <tr> <td>Resource group *</td> <td><input type="text"/></td> </tr> <tr> <td>Virtual network *</td> <td><input type="text" value="Select an option"/></td> </tr> <tr> <td>Subnet *</td> <td><input type="text" value="Select an option"/></td> </tr> <tr> <td>Staging Location *</td> <td><input type="text" value="Select an option"/></td> </tr> </table> <p><a href="#">Can't find your storage account ?</a></p>		Restore Type *	<input type="text" value="Create new virtual machine"/>	Virtual machine name *	<input type="text" value="Enter a name"/>	Resource group *	<input type="text"/>	Virtual network *	<input type="text" value="Select an option"/>	Subnet *	<input type="text" value="Select an option"/>	Staging Location *	<input type="text" value="Select an option"/>
Restore Type *	<input type="text" value="Create new virtual machine"/>												
Virtual machine name *	<input type="text" value="Enter a name"/>												
Resource group *	<input type="text"/>												
Virtual network *	<input type="text" value="Select an option"/>												
Subnet *	<input type="text" value="Select an option"/>												
Staging Location *	<input type="text" value="Select an option"/>												
<b>Restore</b>													

- To restore and create a VM, refer to [Create a VM](#).
- To restore as a disk, refer to [Restore disks](#).

**NOTE**

- You can cancel the restore job till the data transfer phase. Once it enters VM creation phase, you can't cancel the restore job.
- The Cross Region Restore feature restores CMK (customer-managed keys) enabled Azure VMs, which aren't backed-up in a CMK enabled Recovery Services vault, as non-CMK enabled VMs in the secondary region.
- The Azure roles needed to restore in the secondary region are the same as those in the primary region.
- While restoring an Azure VM, Azure Backup configures the virtual network settings in the secondary region automatically. If you are [restoring disks](#) while deploying the template, ensure to provide the virtual network settings, corresponding to the secondary region.
- If VNet/Subnet is not available in the primary region or is not configured in the secondary region, Azure portal doesn't auto-populate any default values during restore operation.
- For Cross Region Restores, the **Staging Location** (that is the storage account location) must be in the region that the Recovery Services vault treats as the *secondary* region. For example, a Recovery Services vault is located in East US 2 region (with Geo-Redundancy and Cross Region Restore enabled). This means that the *secondary* region would be *Central US*. Therefore, you need to create a storage account in *Central US* to perform a Cross Region Restore of the VM.

Learn more about [Azure cross-region replication pairings for all geographies](#).

Azure zone pinned VMs can be restored in any availability zones of the same region.

In the restore process, you'll see the option **Availability Zone**. You'll see your default zone first. To choose a different zone, choose the number of the zone of your choice. If the pinned zone is unavailable, you won't be able to restore the data to another zone because the backed-up data isn't zonally replicated. The restore in

availability zones is possible from recovery points in vault tier only.

In summary, the **Availability Zone** will only appear when

- The source VM is zone pinned and is NOT encrypted
- The recovery point is present in vault tier only (Snapshots only or snapshot and vault tier are not supported)
- The recovery option is to either create a new VM or to restore disks (replace disks option replaces source data and hence the availability zone option is not applicable)
- Creating VM/disks in the same region when vault's storage redundancy is ZRS (Doesn't work when vault's storage redundancy is GRS even though the source VM is zone pinned)
- Creating VM/disks in the paired region when vault's storage redundancy is enabled for Cross-Region-Restore AND if the paired region supports zones

## Restore Virtual Machine

zone1vm

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*

1/26/2021, 12:37:04 AM

Select

Restore Configuration

Create new  Replace existing

**i** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type \* ⓘ

Create new virtual machine

Virtual machine name \* ⓘ

Enter a name

Availability Zone \* ⓘ

1

2

3

Resource group \* ⓘ

Virtual network \* ⓘ

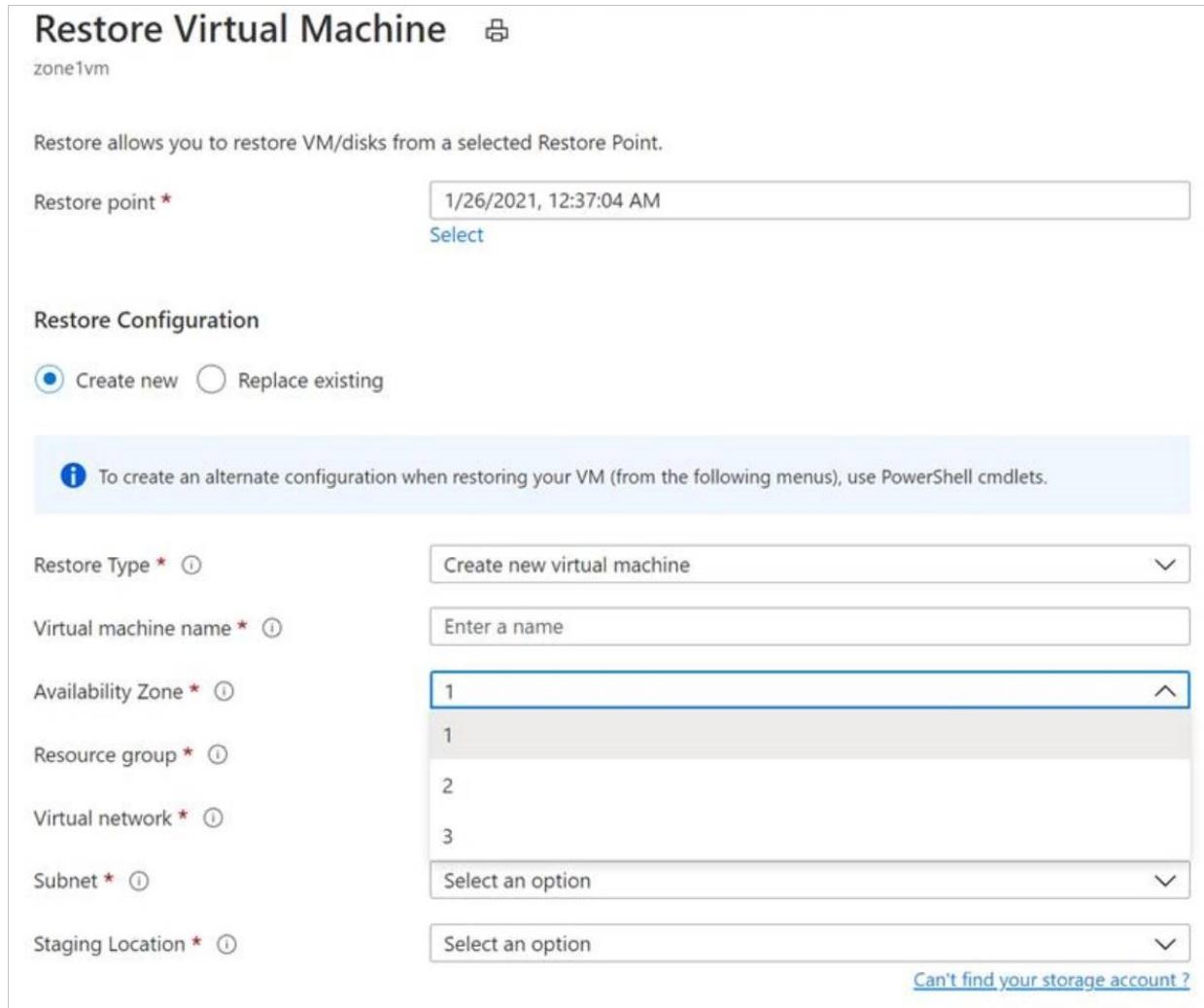
Subnet \* ⓘ

Select an option

Staging Location \* ⓘ

Select an option

[Can't find your storage account ?](#)



### Monitoring secondary region restore jobs

1. From the portal, go to **Recovery Services vault > Backup Jobs**
2. Select **Secondary Region** to view the items in the secondary region.

**Backup jobs**

Showing data from eastus2euap (secondary region)

Filtered by: Item Type - All item types, Operation - All Operations, Status - All Status, Start Time - 12/30/2019, 12:24:51 PM, End Time - 12/31/2019, 12:24:51 PM

Workload name	Operation	Status	Type	Start time	Duration	...
crrignitedemovm	CrossRegionRestore	In progress	Azure virtual machine	12/31/2019, 12:24:37 PM	00:00:15	...
crrignitedemovm	CrossRegionRestore	Completed	Azure virtual machine	12/30/2019, 3:57:29 PM	00:28:53	...

## Restoring unmanaged VMs and disks as managed

You're provided with an option to restore **unmanaged disks** as **managed disks** during restore. By default, the unmanaged VMs / disks are restored as unmanaged VMs / disks. However, if you choose to restore as managed VMs / disks, it's now possible to do so. These restore operations aren't triggered from the snapshot phase but only from the vault phase. This feature isn't available for unmanaged encrypted VMs.

### Restore Virtual Machine

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*  [Select](#)

[Restore as Managed disks/VM](#)

Restore Configuration

Create new  Replace existing

**Tip:** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type \*

Virtual machine name \*

Resource group \*

Virtual network \*

Subnet \*

Staging Location \*

[Can't find your storage account ?](#)

**Restore**

## Restore VMs with special configurations

There are many common scenarios in which you might need to restore VMs.

SCENARIO	GUIDANCE
<b>Restore VMs using Hybrid Use Benefit</b>	If a Windows VM uses <a href="#">Hybrid Use Benefit (HUB) licensing</a> , restore the disks, and create a new VM using the provided template (with <b>License Type</b> set to <b>Windows_Server</b> ), or PowerShell. This setting can also be applied after creating the VM.
<b>Restore VMs during an Azure datacenter disaster</b>	<p>If the vault uses GRS and the primary datacenter for the VM goes down, Azure Backup supports restoring backed-up VMs to the paired datacenter. You select a storage account in the paired datacenter, and restore as normal. Azure Backup uses the compute service in the paired region to create the restored VM. <a href="#">Learn more</a> about datacenter resiliency.</p> <p>If the vault uses GRS, you can choose the new feature, <a href="#">Cross Region Restore</a>. This lets you restore to a second region in either full or partial outage scenarios, or even if there's no outage at all.</p>
<b>Bare-metal restore</b>	The major difference between Azure VMs and on-premises hypervisors is that there's no VM console available in Azure. A console is required for certain scenarios, such as recovering by using a bare-metal recovery (BMR)-type backup. However, VM restore from the vault is a full replacement for BMR.
<b>Restore VMs with special network configurations</b>	Special network configurations include VMs using internal or external load balancing, using multiple NICs, or multiple reserved IP addresses. You restore these VMs by using the <a href="#">restore disk option</a> . This option makes a copy of the VHDs into the specified storage account, and you can then create a VM with an <a href="#">internal</a> or <a href="#">external</a> load balancer, <a href="#">multiple NICs</a> , or <a href="#">multiple reserved IP addresses</a> , in accordance with your configuration.
<b>Network Security Group (NSG) on NIC/Subnet</b>	Azure VM backup supports Backup and Restore of NSG information at vnet, subnet, and NIC level.
<b>Zone Pinned VMs</b>	If you back up an Azure VM that's pinned to a zone (with Azure Backup), then you can restore it in the same zone where it was pinned. <a href="#">Learn more</a>
<b>Restore VM in any availability set</b>	When you restore a VM from the portal, there's no option to choose an availability set. A restored VM doesn't have an availability set. If you use the restore disk option, then you can <a href="#">specify an availability set</a> when you create a VM from the disk using the provided template or PowerShell.
<b>Restore special VMs such as SQL VMs</b>	If you're backing up a SQL VM using Azure VM backup and then use the restore VM option or create a VM after restoring disks, then the newly created VM must be registered with the SQL provider as mentioned <a href="#">here</a> . This will convert the restored VM into a SQL VM.

## Restore domain controller VMs

SCENARIO	GUIDANCE
Restore a single domain controller VM in a single domain	<p>Restore the VM like any other VM. Note that:</p> <p>From an Active Directory perspective, the Azure VM is like any other VM.</p> <p>Directory Services Restore Mode (DSRM) is also available, so all Active Directory recovery scenarios are viable. <a href="#">Learn more</a> about backup and restore considerations for virtualized domain controllers.</p>
Restore multiple domain controller VMs in a single domain	If other domain controllers in the same domain can be reached over the network, the domain controller can be restored like any VM. If it's the last remaining domain controller in the domain, or a recovery in an isolated network is performed, use a <a href="#">forest recovery</a> .
Restore a single domain controller VM in a multiple domain configuration	Restore the disks and create a VM by <a href="#">using PowerShell</a>
Restore multiple domains in one forest	We recommend a <a href="#">forest recovery</a> .

For more information, see [Back up and restore Active Directory domain controllers](#).

## Restore VMs with managed identities

Managed identities eliminate the need for the user to maintain the credentials. Managed identities provide an identity for applications to use when connecting to resources that support Azure Active Directory (Azure AD) authentication.

Azure Backup offers the flexibility to restore the managed Azure VM with [managed identities](#). You can choose to select [system-managed identities](#) or user-managed identities as shown in the figure below. This is introduced as one of the input parameters in the [Restore configuration blade](#) of Azure VM. Managed identities used as one of the input parameters is only used for accessing the storage accounts, which are used as staging location during restore and not for any other Azure resource controlling. These managed identities have to be associated to the vault.

## Restore Virtual Machine

vmhubuntu1804

Restore allows you to restore VM/disks from a selected Restore Point.

Restore point \*

7/27/2021, 3:35:28 PM  
Select

Data Store

Snapshot and Vault-Standard

Restore Configuration

Create new  
 Replace existing

To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type \*

Virtual machine name \*

Resource group \*

Virtual network \*

Subnet \*

Staging Location \*   
Can't find your storage account?

The identities listed here are based on the MSI configurations in the corresponding Recovery services vault. Learn more.

Identities

**Restore**

If you choose to select system-assigned or user-assigned managed identities, check for the below actions for managed identity on the target staging Storage Account.

```
"permissions": [
    {
        "actions": [
            "Microsoft.Authorization/*/read",
            "Microsoft.Storage/storageAccounts/blobServices/containers/delete",
            "Microsoft.Storage/storageAccounts/blobServices/containers/read",
            "Microsoft.Storage/storageAccounts/blobServices/containers/write"
        ],
        "notActions": [],
        "dataActions": [
            "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/delete",
            "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/read",
            "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/write",
            "Microsoft.Storage/storageAccounts/blobServices/containers/blobs/add/action"
        ],
        "notDataActions": []
    }
]
```

Or, add the role assignment on the staging location (Storage Account) to have [Storage account Backup Contributor](#) and [Storage Blob data Contributor](#) for the successful restore operation.

**Restore Virtual Machine**

vmubuntu1804

Restore allows you to restore analysis from a selected restore point.

Restore point \*  Select

Data Store

Restore Configuration

Create new  
 Replace existing

**Note:** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type *	<input type="text" value="Create new virtual machine"/>
Virtual machine name *	<input type="text" value="Enter a name"/>
Resource group *	<input type="text" value="linuxRsVault"/>
Virtual network *	<input type="text" value="Select an option"/>
Subnet *	<input type="text" value="Select an option"/>
Staging Location *	<input type="text" value="Select an option"/>

[Can't find your storage account?](#)

**Note:** The identities listed here are based on the MSI configurations in the corresponding Recovery services vault. Learn more.

Identities  Disabled  
 System assigned  
 User assigned

**Restore**

You can also select the [user-managed identity](#) by providing the input as their MSI Resource ID as provided in the figure below.

Home > linuxRsVault > Backup Items (Azure Virtual Machine) > vmubuntu1804 >

**Restore Virtual Machine**

vmubuntu1804

Restore allows you to restore analysis from a selected restore point.

Restore point \*  Select

Data Store

Restore Configuration

Create new  
 Replace existing

**Note:** To create an alternate configuration when restoring your VM (from the following menus), use PowerShell cmdlets.

Restore Type *	<input type="text" value="Create new virtual machine"/>
Virtual machine name *	<input type="text" value="Enter a name"/>
Resource group *	<input type="text" value="linuxRsVault"/>
Virtual network *	<input type="text" value="Select an option"/>
Subnet *	<input type="text" value="Select an option"/>
Staging Location *	<input type="text" value="Select an option"/>

[Can't find your storage account?](#)

**Note:** The identities listed here are based on the MSI configurations in the corresponding Recovery services vault. Learn more.

Identities  Disabled  
 System assigned  
 User assigned

MSI Resource Id \*

**Restore**

## NOTE

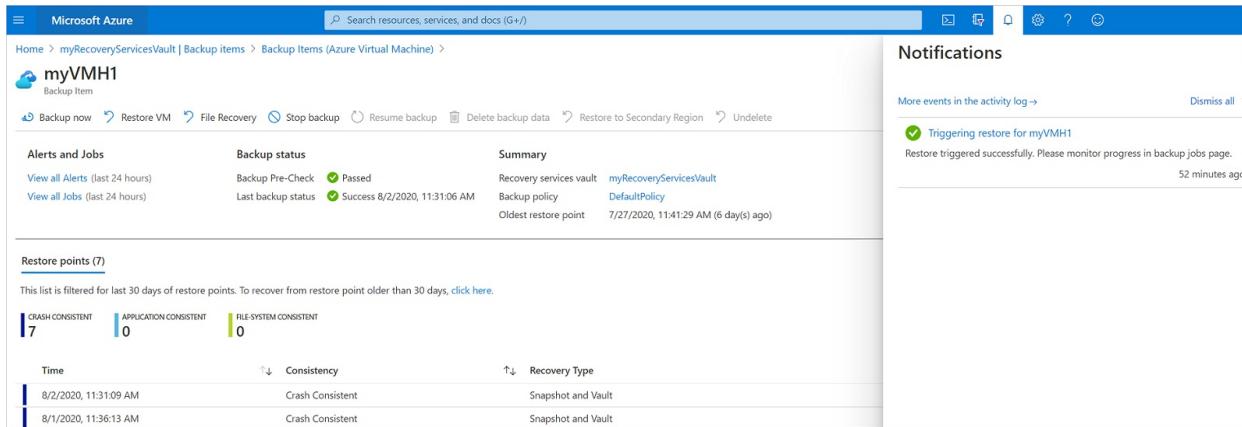
The support is available for only managed VMs, and not supported for classic VMs and unmanaged VMs. For the [storage accounts that are restricted with firewalls](#), system MSI is only supported.

Cross Region Restore isn't supported with managed identities.

Currently, this is available in all Azure public and national cloud regions.

# Track the restore operation

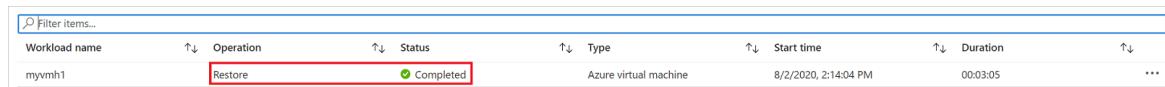
After you trigger the restore operation, the backup service creates a job for tracking. Azure Backup displays notifications about the job in the portal. If they aren't visible, select the **Notifications** symbol, and then select **More events in the activity log** to see the Restore Process Status.



The screenshot shows the Azure portal interface for a backup item named 'myVMH1'. In the top navigation bar, there's a search bar and several icons. Below the navigation, the breadcrumb path is 'Home > myRecoveryServicesVault | Backup items > Backup Items (Azure Virtual Machine) > myVMH1'. Under the 'myVMH1' heading, there are links for 'Backup now', 'Restore VM', 'File Recovery', 'Stop backup', 'Resume backup', 'Delete backup data', 'Restore to Secondary Region', and 'Undelete'. A 'Backup status' section shows 'Backup Pre-Check' as 'Passed' and 'Last backup status' as 'Success 8/2/2020, 11:31:06 AM'. A 'Summary' section indicates the 'Backup policy' is 'DefaultPolicy' and the 'Oldest restore point' is '7/27/2020, 11:41:29 AM (6 day(s) ago)'. Below this, a 'Restore points (7)' section lists two entries: '8/2/2020, 11:31:09 AM' (Crash Consistent, Snapshot and Vault) and '8/1/2020, 11:36:13 AM' (Crash Consistent, Snapshot and Vault). On the right side, the 'Notifications' pane is open, showing a single event: 'Triggering restore for myVMH1' with the message 'Restore triggered successfully. Please monitor progress in backup jobs page.' and a timestamp of '52 minutes ago'. There's also a 'Dismiss all' button.

Track restore as follows:

1. To view operations for the job, select the notifications hyperlink. Alternatively, in the vault, select **Backup jobs**, and then select the relevant VM.



The screenshot shows a table of backup jobs for the 'myvmh1' workload. The columns are: Workload name, Operation, Status, Type, Start time, Duration, and an ellipsis button. One row is highlighted with a red border, showing 'myvmh1' as the workload, 'Restore' as the operation, 'Completed' as the status, 'Azure virtual machine' as the type, '8/2/2020, 2:14:04 PM' as the start time, and '00:03:05' as the duration. The 'Status' column for this row contains a green checkmark icon.

2. To monitor restore progress, select any restore job with a status of **In-progress**. This displays the progress bar, which displays information about the restore progress:

- **Estimated time of restore:** Initially provides the time taken to complete the restore operation. As the operation progresses, the time taken reduces and reaches zero when the restore operation finishes.
- **Percentage of restore:** Shows the percentage of restore operation that's done.
- **Number of bytes transferred:** If you're restoring by creating a new VM, it shows the bytes that were transferred against the total number of bytes to be transferred.

## Post-restore steps

There are a few things to note after restoring a VM:

- Extensions present during the backup configuration are installed, but not enabled. If you see an issue, reinstall the extensions.
- If the backed-up VM had a static IP address, the restored VM will have a dynamic IP address to avoid conflict. You can [add a static IP address to the restored VM](#).
- A restored VM doesn't have an availability set. If you use the restore disk option, then you can [specify an availability set](#) when you create a VM from the disk using the provided template or PowerShell.
- If you use a cloud-init-based Linux distribution, such as Ubuntu, for security reasons the password is blocked after the restore. Use the VMAccess extension on the restored VM to [reset the password](#). We recommend using SSH keys on these distributions, so you don't need to reset the password after the restore.
- If you're unable to access a VM once restored because the VM has a broken relationship with the domain controller, then follow the steps below to bring up the VM:

- Attach OS disk as a data disk to a recovered VM.
- Manually install VM agent if Azure Agent is found to be unresponsive by following this [link](#).
- Enable Serial Console access on VM to allow command-line access to VM

```

bcdedit /store <drive letter>:\boot\bcd /enum
bcdedit /store <VOLUME LETTER WHERE THE BCD FOLDER IS>:\boot\bcd /set {bootmgr} displaybootmenu yes
bcdedit /store <VOLUME LETTER WHERE THE BCD FOLDER IS>:\boot\bcd /set {bootmgr} timeout 5
bcdedit /store <VOLUME LETTER WHERE THE BCD FOLDER IS>:\boot\bcd /set {bootmgr} bootems yes
bcdedit /store <VOLUME LETTER WHERE THE BCD FOLDER IS>:\boot\bcd /ems {<<BOOT LOADER IDENTIFIER>>}
ON
bcdedit /store <VOLUME LETTER WHERE THE BCD FOLDER IS>:\boot\bcd /emssettings EMSPORT:1
EMSBAUDRATE:115200

```

- When the VM is rebuilt use Azure portal to reset local administrator account and password
- Use Serial console access and CMD to disjoin VM from domain

```

cmd /c "netdom remove <> /domain:<> /userD:<> /passwordD:<> /reboot:10 /Force"

```

- Once the VM is disjoined and restarted, you'll be able to successfully RDP to the VM with local admin credentials and rejoin VM back to domain successfully.

## Backing up restored VMs

- If you restored a VM to the same resource group with the same name as the originally backed-up VM, backup continues on the VM after restore.
- If you restored the VM to a different resource group or you specified a different name for the restored VM, you need to set up backup for the restored VM.

## Next steps

- If you experience difficulties during the restore process, [review](#) common issues and errors.
- After the VM is restored, learn about [managing virtual machines](#)

# Support matrix for Azure VM backup

9/21/2022 • 18 minutes to read • [Edit Online](#)

You can use the [Azure Backup service](#) to back up on-premises machines and workloads, and Azure virtual machines (VMs). This article summarizes support settings and limitations when you back up Azure VMs with Azure Backup.

Other support matrices:

- [General support matrix](#) for Azure Backup
- [Support matrix](#) for Azure Backup server / System Center Data Protection Manager (DPM) backup
- [Support matrix](#) for backup with the Microsoft Azure Recovery Services (MARS) agent

## Supported scenarios

Here's how you can back up and restore Azure VMs with the Azure Backup service.

SCENARIO	BACKUP	AGENT	RESTORE
Direct backup of Azure VMs	Back up the entire VM.	No additional agent is needed on the Azure VM. Azure Backup installs and uses an extension to the <a href="#">Azure VM agent</a> that's running on the VM.	<p>Restore as follows:</p> <ul style="list-style-type: none"><li>- <b>Create a basic VM.</b> This is useful if the VM has no special configuration such as multiple IP addresses.</li><li>- <b>Restore the VM disk.</b> Restore the disk. Then attach it to an existing VM, or create a new VM from the disk by using PowerShell.</li><li>- <b>Replace VM disk.</b> If a VM exists and it uses managed disks (unencrypted), you can restore a disk and use it to replace an existing disk on the VM.</li><li>- <b>Restore specific files/folders.</b> You can restore files/folders from a VM instead of from the entire VM.</li></ul>
Direct backup of Azure VMs (Windows only)	Back up specific files/folders/volume.	Install the <a href="#">Azure Recovery Services agent</a> .  You can run the MARS agent alongside the backup extension for the Azure VM agent to back up the VM at file/folder level.	Restore specific folders/files.

SCENARIO	BACKUP	AGENT	RESTORE
Back up Azure VM to the backup server	Back up files/folders/volumes; system state/bare metal files; app data to System Center DPM or to Microsoft Azure Backup Server (MABS).  DPM/MABS then backs up to the backup vault.	Install the DPM/MABS protection agent on the VM. The MARS agent is installed on DPM/MABS.	Restore files/folders/volumes; system state/bare metal files; app data.

Learn more about backup [using a backup server](#) and about [support requirements](#).

## Supported backup actions

ACTION	SUPPORT
Back up a VM that's shutdown/offline VM	Supported.  Snapshot is crash-consistent only, not app-consistent.
Back up disks after migrating to managed disks	Supported.  Backup will continue to work. No action is required.
Back up managed disks after enabling resource group lock	Not supported.  Azure Backup can't delete the older restore points, and backups will start to fail when the maximum limit of restore points is reached.
Modify backup policy for a VM	Supported.  The VM will be backed up by using the schedule and retention settings in new policy. If retention settings are extended, existing recovery points are marked and kept. If they're reduced, existing recovery points will be pruned in the next cleanup job and eventually deleted.
Cancel a backup job	Supported during snapshot process.  Not supported when the snapshot is being transferred to the vault.
Back up the VM to a different region or subscription	Not supported.  To successfully back up, virtual machines must be in the same subscription as the vault for backup.
Backups per day (via the Azure VM extension)	Four backups per day - one scheduled backup as per the Backup policy, and three on-demand backups.  However, to allow user retries in case of failed attempts, hard limit for on-demand backups is set to nine attempts.
Backups per day (via the MARS agent)	Three scheduled backups per day.

ACTION	SUPPORT
Backups per day (via DPM/MABS)	Two scheduled backups per day.
Monthly/yearly backup	<p>Not supported when backing up with Azure VM extension. Only daily and weekly is supported.</p> <p>You can set up the policy to retain daily/weekly backups for monthly/yearly retention period.</p>
Automatic clock adjustment	<p>Not supported.</p> <p>Azure Backup doesn't automatically adjust for daylight saving time changes when backing up a VM.</p> <p>Modify the policy manually as needed.</p>
<a href="#">Security features for hybrid backup</a>	Disabling security features isn't supported.
Back up the VM whose machine time is changed	<p>Not supported.</p> <p>If the machine time is changed to a future date-time after enabling backup for that VM, however even if the time change is reverted, successful backup isn't guaranteed.</p>
Multiple Backups Per Day	<p>Supported (in preview), using <i>Enhanced policy</i> (in preview).</p> <p>For hourly backup, the minimum RPO is 4 hours and the maximum is 24 hours. You can set the backup schedule to 4, 6, 8, 12, and 24 hours respectively. Learn about how to <a href="#">back up an Azure VM using Enhanced policy</a>.</p>
Back up a VM with deprecated plan when publisher has removed it from Azure Marketplace	<p>Not supported.</p> <p>Backup is possible. However, restore will fail.</p> <p>If you've already configured backup for VM with deprecated virtual machine offer and encounter restore error, see <a href="#">Troubleshoot backup errors with Azure VMs</a>.</p>

## Operating system support (Windows)

The following table summarizes the supported operating systems when backing up Azure VMs running Windows.

SCENARIO	OS SUPPORT

SCENARIO	OS SUPPORT
Back up with Azure VM agent extension	<ul style="list-style-type: none"> <li>- Windows 10 Client (64 bit only)</li> <li>- Windows Server 2022 (Datacenter/Datacenter Core/Standard)</li> <li>- Windows Server 2019 (Datacenter/Datacenter Core/Standard)</li> <li>- Windows Server 2016 (Datacenter/Datacenter Core/Standard)</li> <li>- Windows Server 2012 R2 (Datacenter/Standard)</li> <li>- Windows Server 2012 (Datacenter/Standard)</li> <li>- Windows Server 2008 R2 (RTM and SP1 Standard)</li> <li>- Windows Server 2008 (64 bit only)</li> </ul>
Back up with MARS agent	<a href="#">Supported</a> operating systems.
Back up with DPM/MABS	Supported operating systems for backup with <a href="#">MABS</a> and <a href="#">DPM</a> .

Azure Backup doesn't support 32-bit operating systems.

## Support for Linux backup

Here's what's supported if you want to back up Linux machines.

ACTION	SUPPORT
Back up Linux Azure VMs with the Linux Azure VM agent	<p>File consistent backup.</p> <p>App-consistent backup using <a href="#">custom scripts</a>.</p> <p>During restore, you can create a new VM, restore a disk and use it to create a VM, or restore a disk, and use it to replace a disk on an existing VM. You can also restore individual files and folders.</p>
Back up Linux Azure VMs with MARS agent	<p>Not supported.</p> <p>The MARS agent can only be installed on Windows machines.</p>
Back up Linux Azure VMs with DPM/MABS	Not supported.
Back up Linux Azure VMs with docker mount points	Currently, Azure Backup doesn't support exclusion of docker mount points as these are mounted at different paths every time.

## Operating system support (Linux)

For Azure VM Linux backups, Azure Backup supports the list of Linux [distributions endorsed by Azure](#). Note the following:

- Azure Backup doesn't support Core OS Linux.
- Azure Backup doesn't support 32-bit operating systems.
- Other bring-your-own Linux distributions might work as long as the [Azure VM agent for Linux](#) is available on the VM, and as long as Python is supported.
- Azure Backup doesn't support a proxy-configured Linux VM if it doesn't have Python version 2.7 or higher installed.
- Azure Backup doesn't support backing up NFS files that are mounted from storage, or from any other NFS server, to Linux or Windows machines. It only backs up disks that are locally attached to the VM.

## Support matrix for managed pre-post scripts for Linux databases

Azure Backup provides support for customers to author their own pre-post scripts

SUPPORTED DATABASE	OS VERSION	DATABASE VERSION
Oracle in Azure VMs	Oracle Linux	Oracle 12.x or greater

## Backup frequency and retention

SETTING	LIMITS
Maximum recovery points per protected instance (machine/workload)	9999.
Maximum expiry time for a recovery point	No limit (99 years).
Maximum backup-frequency to vault (Azure VM extension)	Once a day.
Maximum backup-frequency to vault (MARS agent)	Three backups per day.
Maximum backup-frequency to DPM/MABS	Every 15 minutes for SQL Server. Once an hour for other workloads.
Recovery point retention	Daily, weekly, monthly, and yearly.
Maximum retention period	Depends on backup frequency.
Recovery points on DPM/MABS disk	64 for file servers, and 448 for app servers. Tape recovery points are unlimited for on-premises DPM.

## Supported restore methods

RESTORE OPTION	DETAILS
Create a new VM	Quickly creates and gets a basic VM up and running from a restore point.  You can specify a name for the VM, select the resource group and virtual network (VNet) in which it will be placed, and specify a storage account for the restored VM. The new VM must be created in the same region as the source VM.

RESTORE OPTION	DETAILS
<b>Restore disk</b>	<p>Restores a VM disk, which can then be used to create a new VM.</p> <p>Azure Backup provides a template to help you customize and create a VM.</p> <p>The restore job generates a template that you can download and use to specify custom VM settings, and create a VM.</p> <p>The disks are copied to the Resource Group you specify.</p> <p>Alternatively, you can attach the disk to an existing VM, or create a new VM using PowerShell.</p> <p>This option is useful if you want to customize the VM, add configuration settings that weren't there at the time of backup, or add settings that must be configured using the template or PowerShell.</p>
<b>Replace existing</b>	<p>You can restore a disk, and use it to replace a disk on the existing VM.</p> <p>The current VM must exist. If it's been deleted, this option can't be used.</p> <p>Azure Backup takes a snapshot of the existing VM before replacing the disk, and stores it in the staging location you specify. Existing disks connected to the VM are replaced with the selected restore point.</p> <p>The snapshot is copied to the vault, and retained in accordance with the retention policy.</p> <p>After the replace disk operation, the original disk is retained in the resource group. You can choose to manually delete the original disks if they aren't needed.</p> <p>Replace existing is supported for unencrypted managed VMs and for VMs <a href="#">created using custom images</a>. It's not supported for unmanaged disks and VMs, classic VMs, and <a href="#">generalized VMs</a>.</p> <p>If the restore point has more or less disks than the current VM, then the number of disks in the restore point will only reflect the VM configuration.</p> <p>Replace existing is also supported for VMs with linked resources, like <a href="#">user-assigned managed-identity</a> and <a href="#">Key Vault</a>.</p>

RESTORE OPTION	DETAILS
Cross Region (secondary region)	<p>Cross Region restore can be used to restore Azure VMs in the secondary region, which is an <a href="#">Azure paired region</a>.</p> <p>You can restore all the Azure VMs for the selected recovery point if the backup is done in the secondary region.</p> <p>This feature is available for the options below:</p> <ul style="list-style-type: none"> <li>• <a href="#">Create a VM</a></li> <li>• <a href="#">Restore Disks</a></li> </ul> <p>We don't currently support the <a href="#">Replace existing disks</a> option.</p> <p>Permissions The restore operation on secondary region can be performed by Backup Admins and App admins.</p>

## Support for file-level restore

RESTORE	SUPPORTED
Restoring files across operating systems	You can restore files on any machine that has the same (or compatible) OS as the backed-up VM. See the <a href="#">Compatible OS table</a> .
Restoring files from encrypted VMs	Not supported.
Restoring files from network-restricted storage accounts	Not supported.
Restoring files on VMs using Windows Storage Spaces	Restore not supported on same VM.  Instead, restore the files on a compatible VM.
Restore files on Linux VM using LVM/raid arrays	Restore not supported on same VM.  Restore on a compatible VM.
Restore files with special network settings	Restore not supported on same VM.  Restore on a compatible VM.
Restore files from Shared disk, Temp drive, Deduplicated Disk, Ultra disk and disk with write Accelerator enabled	Restore not supported, see <a href="#">Azure VM storage support</a> .

## Support for VM management

The following table summarizes support for backup during VM management tasks, such as adding or replacing VM disks.

RESTORE	SUPPORTED
Restore across subscription	<a href="#">Cross Subscription Restore</a> is now supported in Azure VMs.
<a href="#">Restore across region</a>	Supported.

RESTORE	SUPPORTED
Restore across zone	Unsupported.
Restore to an existing VM	Use replace disk option.
Restore disk with storage account enabled for Azure Storage Service Encryption (SSE)	Not supported.  Restore to an account that doesn't have SSE enabled.
Restore to mixed storage accounts	Not supported.  Based on the storage account type, all restored disks will be either premium or standard, and not mixed.
Restore VM directly to an availability set	For managed disks, you can restore the disk and use the availability set option in the template.  Not supported for unmanaged disks. For unmanaged disks, restore the disk, and then create a VM in the availability set.
Restore backup of unmanaged VMs after upgrading to managed VM	Supported.  You can restore disks, and then create a managed VM.
Restore VM to restore point before the VM was migrated to managed disks	Supported.  You restore to unmanaged disks (default), convert the restored disks to managed disk, and create a VM with the managed disks.
Restore a VM that's been deleted.	Supported.  You can restore the VM from a recovery point.
Restore a domain controller VM	Supported. For details, see <a href="#">Restore domain controller VMs</a> .
Restore VM in different virtual network	Supported.  The virtual network must be in the same subscription and region.

## VM compute support

COMPUTE	SUPPORT
VM size	Any Azure VM size with at least 2 CPU cores and 1-GB RAM.  <a href="#">Learn more</a> .
Back up VMs in <a href="#">availability sets</a>	Supported.  You can't restore a VM in an available set by using the option to quickly create a VM. Instead, when you restore the VM, restore the disk and use it to deploy a VM, or restore a disk and use it to replace an existing disk.

COMPUTE	SUPPORT
Back up VMs that are deployed with <a href="#">Hybrid Use Benefit (HUB)</a>	Supported.
Back up VMs that are deployed from <a href="#">Azure Marketplace</a> (Published by Microsoft, third party)	Supported.  The VM must be running a supported operating system.  When recovering files on the VM, you can restore only to a compatible OS (not an earlier or later OS). We don't restore Azure Marketplace VMs backed as VMs, as these need purchase information. They're only restored as disks.
Back up VMs that are deployed from a custom image (third-party)	Supported.  The VM must be running a supported operating system.  When recovering files on the VM, you can restore only to a compatible OS (not an earlier or later OS).
Back up VMs that are migrated to Azure	Supported.  To back up the VM, the VM agent must be installed on the migrated machine.
Back up Multi-VM consistency	Azure Backup doesn't provide data and application consistency across multiple VMs.
Backup with <a href="#">Diagnostic Settings</a>	Unsupported.  If the restore of the Azure VM with diagnostic settings is triggered using the <a href="#">Create New</a> option, then the restore fails.
Restore of Zone-pinned VMs	Supported (for a VM that's backed-up after Jan 2019 and where <a href="#">availability zones</a> are available).  We currently support restoring to the same zone that's pinned in VMs. However, if the zone is unavailable due to an outage, the restore will fail.
Gen2 VMs	Supported Azure Backup supports backup and restore of <a href="#">Gen2 VMs</a> . When these VMs are restored from Recovery point, they're restored as <a href="#">Gen2 VMs</a> .
Backup of Azure VMs with locks	Unsupported for unmanaged VMs.  Supported for managed VMs.
Spot VMs	Unsupported. Azure Backup restores Spot VMs as regular Azure VMs.

COMPUTE	SUPPORT
Azure Dedicated Host	<p>Supported</p> <p>While restoring an Azure VM through the <a href="#">Create New</a> option, though the restore gets successful, Azure VM can't be restored in the dedicated host. To achieve this, we recommend you to restore as disks. While <a href="#">restoring as disks</a> with the template, create a VM in dedicated host, and then attach the disks.</p> <p>This is not applicable in secondary region, while performing <a href="#">Cross Region Restore</a>.</p>
Windows Storage Spaces configuration of standalone Azure VMs	Supported
Azure Virtual Machine Scale Sets	Supported for flexible orchestration model to back up and restore Single Azure VM.
Restore with Managed identities	<p>Yes, supported for managed Azure VMs, and not supported for classic and unmanaged Azure VMs.</p> <p>Cross Region Restore isn't supported with managed identities.</p> <p>Currently, this is available in all Azure public and national cloud regions.</p> <p><a href="#">Learn more</a>.</p>
Trusted Launch VM	<p>Backup supported.</p> <p>Backup of Trusted Launch VM is supported through <a href="#">Enhanced policy</a>. You can enable backup through <a href="#">Recovery Services vault</a>, <a href="#">VM Manage blade</a>, and <a href="#">Create VM blade</a>.</p> <p><b>Feature details</b></p> <ul style="list-style-type: none"> <li>• Backup is supported in all regions where Trusted Launch VM is available.</li> <li>• Configurations of Backup, Alerts, and Monitoring for Trusted Launch VM are currently not supported through Backup center.</li> <li>• Migration of an existing <a href="#">Generation 2</a> VM (protected with Azure Backup) to Trusted Launch VM is currently not supported. Learn about how to <a href="#">create a Trusted Launch VM</a>.</li> </ul>

## VM storage support

COMPONENT	SUPPORT
Azure VM data disks	<p>Support for backup of Azure VMs with up to 32 disks.</p> <p>Support for backup of Azure VMs with unmanaged disks or classic VMs is up to 16 disks only.</p>

Component	Support
Data disk size	Individual disk size can be up to 32 TB and a maximum of 256 TB combined for all disks in a VM.
Storage type	Standard HDD, Standard SSD, Premium SSD.  Backup and restore of <a href="#">ZRS disks</a> is supported.
Managed disks	Supported.
Encrypted disks	Supported.  Azure VMs enabled with Azure Disk Encryption can be backed up (with or without the Azure AD app).  Encrypted VMs can't be recovered at the file/folder level. You must recover the entire VM.  You can enable encryption on VMs that are already protected by Azure Backup.
Disks with Write Accelerator enabled	Azure VM with WA disk backup is available in all Azure public regions starting from May 18, 2020. If WA disk backup is not required as part of VM backup, you can choose to remove with <a href="#">Selective disk feature</a> .  <b>Important</b> Virtual machines with WA disks need internet connectivity for a successful backup (even though those disks are excluded from the backup).
Disks enabled for access with private EndPoint	Unsupported.
Back up & Restore deduplicated VMs/disks	Azure Backup doesn't support deduplication. For more information, see this <a href="#">article</a> <ul style="list-style-type: none"> <li>- Azure Backup doesn't deduplicate across VMs in the Recovery Services vault</li> <li>- If there are VMs in deduplication state during restore, the files can't be restored because the vault doesn't understand the format. However, you can successfully perform the full VM restore.</li> </ul>
Add disk to protected VM	Supported.
Resize disk on protected VM	Supported.
Shared storage	Backing up VMs using Cluster Shared Volume (CSV) or Scale-Out File Server isn't supported. CSV writers are likely to fail during backup. On restore, disks containing CSV volumes might not come-up.
Shared disks	Not supported.
Ultra SSD disks	Not supported. For more information, see these <a href="#">limitations</a> .

Component	Support
Temporary disks	Temporary disks aren't backed up by Azure Backup.
NVMe/ephemeral disks	Not supported.
ReFS restore	Supported. VSS supports app-consistent backups on ReFS also like NFS.
Dynamic disk with spanned/stripped volumes	<p>Supported</p> <p>If you enable selective disk feature on an Azure VM, then this won't be supported.</p>

## VM network support

Component	Support
Number of network interfaces (NICs)	<p>Up to maximum number of NICs supported for a specific Azure VM size.</p> <p>NICs are created when the VM is created during the restore process.</p> <p>The number of NICs on the restored VM mirrors the number of NICs on the VM when you enabled protection. Removing NICs after you enable protection doesn't affect the count.</p>
External/internal load balancer	<p>Supported.</p> <p><a href="#">Learn more</a> about restoring VMs with special network settings.</p>
Multiple reserved IP addresses	<p>Supported.</p> <p><a href="#">Learn more</a> about restoring VMs with special network settings.</p>
VMs with multiple network adapters	<p>Supported.</p> <p><a href="#">Learn more</a> about restoring VMs with special network settings.</p>
VMs with public IP addresses	<p>Supported.</p> <p>Associate an existing public IP address with the NIC, or create an address and associate it with the NIC after restore is done.</p>
Network security group (NSG) on NIC/subnet.	Supported.

COMPONENT	SUPPORT
Static IP address	<p>Not supported.</p> <p>A new VM that's created from a restore point is assigned a dynamic IP address.</p> <p>For classic VMs, you can't back up a VM with a reserved IP address and no defined endpoint.</p>
Dynamic IP address	<p>Supported.</p> <p>If the NIC on the source VM uses dynamic IP addressing, by default the NIC on the restored VM will use it too.</p>
Azure Traffic Manager	<p>Supported.</p> <p>If the backed-up VM is in Traffic Manager, manually add the restored VM to the same Traffic Manager instance.</p>
Azure DNS	Supported.
Custom DNS	Supported.
Outbound connectivity via HTTP proxy	<p>Supported.</p> <p>An authenticated proxy isn't supported.</p>
Virtual network service endpoints	<p>Supported.</p> <p>Firewall and virtual network storage account settings should allow access from all networks.</p>

## VM security and encryption support

Azure Backup supports encryption for in-transit and at-rest data:

Network traffic to Azure:

- Backup-traffic from servers to the Recovery Services vault is encrypted by using Advanced Encryption Standard 256.
- Backup data is sent over a secure HTTPS link.
- The backup data is stored in the Recovery Services vault in encrypted form.
- Only you have the encryption key to unlock this data. Microsoft can't decrypt the backup data at any point.

### WARNING

After you set up the vault, only you have access to the encryption key. Microsoft never maintains a copy and doesn't have access to the key. If the key is misplaced, Microsoft can't recover the backup data.

Data security:

- When backing up Azure VMs, you need to set up encryption *within* the virtual machine.

- Azure Backup supports Azure Disk Encryption, which uses BitLocker on virtual machines running Windows and uses **dm-crypt** on Linux virtual machines.
- On the back end, Azure Backup uses [Azure Storage Service encryption](#), which protects data at rest.

MACHINE	IN TRANSIT	AT REST
On-premises Windows machines without DPM/MABS		
Azure VMs		
On-premises/Azure VMs with DPM		
On-premises/Azure VMs with MABS		

## VM compression support

Backup supports the compression of backup traffic, as summarized in the following table. Note the following:

- For Azure VMs, the VM extension reads the data directly from the Azure storage account over the storage network. It isn't necessary to compress this traffic.
- If you're using DPM or MABS, you can save bandwidth by compressing the data before it's backed up to DPM/MABS.

MACHINE	COMPRESS TO MABS/DPM (TCP)	COMPRESS TO VAULT (HTTPS)
On-premises Windows machines without DPM/MABS	NA	
Azure VMs	NA	NA
On-premises/Azure VMs with DPM		
On-premises/Azure VMs with MABS		

## Next steps

- [Back up Azure VMs](#).
- [Back up Windows machines directly](#), without a backup server.
- [Set up MABS](#) for backup to Azure, and then back up workloads to MABS.
- [Set up DPM](#) for backup to Azure, and then back up workloads to DPM.

# Azure Backup pricing

9/21/2022 • 8 minutes to read • [Edit Online](#)

To learn about Azure Backup pricing, visit the [Azure Backup pricing page](#).

## Download detailed estimates for Azure Backup pricing

If you're looking to estimate your costs for budgeting or cost comparison purposes, download the detailed [Azure Backup pricing estimator](#).

### What does the estimator contain?

The Azure Backup cost estimator sheet has an option for you to estimate all possible workloads you're looking to back up using Azure Backup. These workloads include:

- Azure VMs
- On-premises servers
- SQL in Azure VMs
- SAP HANA in Azure VMs
- Azure files shares

## Estimate costs for backing up Azure VMs or on-premises servers

To estimate the costs of backing up Azure VMs or on-premises servers using Azure Backup, you'll need the following parameters:

- Size of the VMs or on-premises servers that you're trying to back up
  - Enter the "used size" of disks or servers required to be backed up
- Number of servers with that size
- What is the expected amount of data churn on these servers?

Churn refers to the amount of change in data. For example, if you had a VM with 200 GB of data to be backed up and 10 GB of it changes every day, the daily churn is 5%.

  - Higher churn will mean that you back up more data
  - Pick **Low** or **Moderate** for file servers and **High** if you're running databases
  - If you know your **churn%**, you can use the **Enter your own%** option
- Choose the backup policy
  - How long do you expect to retain "Daily" backups? (in days)
  - How long do you expect to retain "Weekly" backups? (in weeks)
  - How long do you expect to retain "Monthly" backups? (in months)
  - How long do you expect to retain "Yearly" backups? (in years)
  - How long do you expect to retain "Instant restore snapshots"? (1-5 days)
  - This option lets you restore from as far back as seven days in a quick manner using snapshots stored on disks.

- **Optional** – Selective Disk backup
  - If you're using the **Selective Disk Backup** option while backing up Azure VMs, choose the **Exclude Disk** option and enter the percentage of disks excluded from backup in terms of size. For example, if you have a VM connected to three disks with 200 GB used in each disk and if you want to exclude two of them from backing up, enter 66.7%.
- **Optional** – Backup Storage Redundancy
  - This indicates the redundancy of the Storage Account your backup data goes into. We recommend using **GRS** for the highest availability. Since it ensures that a copy of your backup data is kept in a different region, it helps you meet multiple compliance standards. Change the redundancy to **LRS** if you're backing up development or test environments that don't need an enterprise-level backup. Select the **RAGRS** option in the sheet if you want to understand costs when **Cross-Region Restore** is enabled for your backups.
- **Optional** – Modify regional pricing or apply discounted rates
  - If you want to check your estimates for a different region or discounted rates, select **Yes** for the **Try estimates for a different region?** option and enter the rates with which you want to run the estimates.

## Estimate costs for backing up SQL servers in Azure VMs

To estimate the costs of backing up SQL servers running in Azure VMs using Azure Backup, you'll need the following parameters:

- Size of the SQL servers that you're trying to back up
- Number of SQL servers with the above size
- What is the expected compression for your SQL servers' backup data?
  - Most Azure Backup customers see that the backup data has 80% compression compared to the SQL server size when the SQL compression is **enabled**.
  - If you expect to see a different compression, enter the number in this field
- What is the expected size of log backups?
  - The % indicates daily log size as a % of the SQL server size
- What is the expected amount of daily data churn on these servers?
  - Typically, databases have "High" churn
  - If you know your **churn%**, you can use the **Enter your own%** option
- Choose the backup policy
  - Backup Type
    - The most effective policy you can choose is **Daily differentials** with weekly/monthly/yearly full backups. Azure Backup can restore from differentials through single-click as well.
    - You can also choose to have a policy with daily/weekly/monthly/yearly full backups. This option will consume slightly more storage than the first option.
    - How long do you expect to retain "log" backups? (in days) [7-35]
    - How long do you expect to retain "Daily" backups? (in days)

- How long do you expect to retain "Weekly" backups? (in weeks)
- How long do you expect to retain "Monthly" backups? (in months)
- How long do you expect to retain "Yearly" backups? (in years)
- **Optional – Backup Storage Redundancy**
  - This indicates the redundancy of the Storage Account your backup data goes into. We recommend using **GRS** for the highest availability. Since it ensures that a copy of your backup data is kept in a different region, it helps you meet multiple compliance standards. Change the redundancy to **LRS** if you're backing up development or test environments that don't need an enterprise-level backup.
- **Optional – Modify regional pricing or apply discounted rates**
  - If you want to check your estimates for a different region or discounted rates, select **Yes** for the **Try estimates for a different region?** option and enter the rates with which you want to run the estimates.

## Estimate costs for backing up SAP HANA servers in Azure VMs

To estimate the costs of backing up SAP HANA servers running in Azure VMs using Azure Backup, you'll need the following parameters:

- Total size of the SAP HANA databases that you're trying to back up. This should be the sum of full backup size of each of the databases, as reported by SAP HANA.
- Number of SAP HANA servers with the above size
- What is the expected size of log backups?
  - The % indicates average daily log size as a % of the total size of SAP HANA databases that you're backing up on the SAP HANA server
- What is the expected amount of daily data churn on these servers?
  - The % indicates average daily churn size as a % of the total size of SAP HANA databases that you're backing up on the SAP HANA server
  - Typically, databases have "High" churn
  - If you know your **churn%**, you can use the **Enter your own%** option
- Choose the backup policy
  - Backup Type
    - The most effective policy you can choose is **Daily differentials** with **weekly/monthly/yearly** full backups. Azure Backup can restore from differentials through single-click as well.
    - You can also choose to have a policy with **daily/weekly/monthly/yearly** full backups. This option will consume slightly more storage than the first option.
  - How long do you expect to retain "log" backups? (in days) [7-35]
  - How long do you expect to retain "Daily" backups? (in days)
  - How long do you expect to retain "Weekly" backups? (in weeks)
  - How long do you expect to retain "Monthly" backups? (in months)
  - How long do you expect to retain "Yearly" backups? (in years)
- **Optional – Backup Storage Redundancy**
  - This indicates the redundancy of the Storage Account your backup data goes into. We recommend using **GRS** for the highest availability. Since it ensures that a copy of your backup data is kept in a different region, it helps you meet multiple compliance standards. Change the redundancy to **LRS** if

you're backing up development or test environments that don't need an enterprise-level backup.

- **Optional** – Modify regional pricing or apply discounted rates
  - If you want to check your estimates for a different region or discounted rates, select **Yes** for the **Try estimates for a different region?** option and enter the rates with which you want to run the estimates.

## Estimate costs for backing up Azure file shares

To estimate the costs of backing up Azure file shares using the [snapshot-based backup solution](#) offered by Azure Backup, you'll need the following parameters:

- Size (in GB) of the file shares that you want to back up.
- If you want to back up file shares spread across multiple storage accounts, specify the number of storage accounts hosting the file shares with the above size.
- Expected amount of data churn on the file shares that you want to back up.

Churn refers to the amount of change in data and it directly impacts the snapshot storage size. For example, if you have a file share with 200 GB of data to be backed up, and 10 GB of it changes every day, the daily churn is 5%.

  - Higher churn means the amount of data change in the file share contents every day is high, and so incremental snapshot (capturing only the data changes) size would also be more.
  - Select Low (1%), Moderate (3%), or High (5%) based on your file share characteristics and usage.
  - If you know the exact **churn%** for your file share, you can select the **Enter your own%** option from the drop-down. Specify the values (in %) for daily, weekly, monthly, and yearly churn.
- Type of storage account (standard or premium) and the storage redundancy setting of the storage account hosting the backed-up file share.

In the current backup solution for Azure file shares, snapshots are stored in the same storage account as the backed-up file share. So the storage cost associated with snapshots is billed as part of your Azure files bill, based on the snapshot pricing for the account type and redundancy setting of the storage account hosting the backed-up file share and snapshots.
- Retention for different backups
  - How long do you expect to retain "Daily" backups? (in days)
  - How long do you expect to retain "Weekly" backups? (in weeks)
  - How long do you expect to retain "Monthly" backups? (in months)
  - How long do you expect to retain "Yearly" backups? (in years)

Refer to [the Azure File share support matrix](#) for the maximum supported retention values in each category.
- **Optional** – Modify regional pricing or apply discounted rates.
  - The default values set for snapshot storage cost per GB and protected instance cost in the estimator are for the East US region. If you want to check your estimates for a different region or discounted rates, select **Yes** for the **Try estimates for a different region?** option, and enter the rates with which you want to run the estimates.

## Next steps

[What is the Azure Backup service?](#)



# About Site Recovery

9/21/2022 • 3 minutes to read • [Edit Online](#)

Welcome to the Azure Site Recovery service! This article provides a quick service overview.

As an organization, you need to adopt a business continuity and disaster recovery (BCDR) strategy that keeps your data safe, and your apps and workloads online, when planned and unplanned outages occur.

Azure Recovery Services contributes to your BCDR strategy:

- **Site Recovery service:** Site Recovery helps ensure business continuity by keeping business apps and workloads running during outages. Site Recovery [replicates](#) workloads running on physical and virtual machines (VMs) from a primary site to a secondary location. When an outage occurs at your primary site, you fail over to a secondary location, and access apps from there. After the primary location is running again, you can fail back to it.
- **Backup service:** The [Azure Backup](#) service keeps your data safe and recoverable.

Site Recovery can manage replication for:

- Azure VMs replicating between Azure regions.
- On-premises VMs, Azure Stack VMs, and physical servers.

## What does Site Recovery provide?

FEATURE	DETAILS
Simple BCDR solution	Using Site Recovery, you can set up and manage replication, failover, and fallback from a single location in the Azure portal.
Azure VM replication	You can set up disaster recovery of Azure VMs from a primary region to a secondary region.
VMware VM replication	You can replicate VMware VMs to Azure using the improved Azure Site Recovery replication appliance that offers better security and resilience than the configuration server. For more information, see <a href="#">Disaster recovery of VMware VMs</a> .
On-premises VM replication	You can replicate on-premises VMs and physical servers to Azure, or to a secondary on-premises datacenter. Replication to Azure eliminates the cost and complexity of maintaining a secondary datacenter.
Workload replication	Replicate any workload running on supported Azure VMs, on-premises Hyper-V and VMware VMs, and Windows/Linux physical servers.
Data resilience	Site Recovery orchestrates replication without intercepting application data. When you replicate to Azure, data is stored in Azure storage, with the resilience that provides. When failover occurs, Azure VMs are created based on the replicated data.

FEATURE	DETAILS
RTO and RPO targets	Keep recovery time objectives (RTO) and recovery point objectives (RPO) within organizational limits. Site Recovery provides continuous replication for Azure VMs and VMware VMs, and replication frequency as low as 30 seconds for Hyper-V. You can reduce RTO further by integrating with <a href="#">Azure Traffic Manager</a> .
Keep apps consistent over failover	You can replicate using recovery points with application-consistent snapshots. These snapshots capture disk data, all data in memory, and all transactions in process.
Testing without disruption	You can easily run disaster recovery drills, without affecting ongoing replication.
Flexible failovers	You can run planned failovers for expected outages with zero-data loss. Or, unplanned failovers with minimal data loss, depending on replication frequency, for unexpected disasters. You can easily fail back to your primary site when it's available again.
Customized recovery plans	Using recovery plans, you can customize and sequence the failover and recovery of multi-tier applications running on multiple VMs. You group machines together in a recovery plan, and optionally add scripts and manual actions. Recovery plans can be integrated with Azure automation runbooks.
BCDR integration	Site Recovery integrates with other BCDR technologies. For example, you can use Site Recovery to protect the SQL Server backend of corporate workloads, with native support for SQL Server Always On, to manage the failover of availability groups.
Azure automation integration	A rich Azure Automation library provides production-ready, application-specific scripts that can be downloaded and integrated with Site Recovery.
Network integration	Site Recovery integrates with Azure for application network management. For example, to reserve IP addresses, configure load-balancers, and use Azure Traffic Manager for efficient network switchovers.

## What can I replicate?

SUPPORTED	DETAILS
Replication scenarios	<ul style="list-style-type: none"> <li>Replicate Azure VMs from one Azure region to another.</li> <li>Replicate on-premises VMware VMs, Hyper-V VMs, physical servers (Windows and Linux), Azure Stack VMs to Azure.</li> <li>Replicate AWS Windows instances to Azure.</li> <li>Replicate on-premises VMware VMs, Hyper-V VMs managed by System Center VMM, and physical servers to a secondary site.</li> </ul>

SUPPORTED	DETAILS
<b>Regions</b>	Review <a href="#">supported regions</a> for Site Recovery.
<b>Replicated machines</b>	Review the replication requirements for <a href="#">Azure VM replication</a> , <a href="#">on-premises VMware VMs</a> and physical servers, and <a href="#">on-premises Hyper-V VMs</a> .
<b>Workloads</b>	You can replicate any workload running on a machine that's supported for replication. And, the Site Recovery team did app-specific tests for a <a href="#">number of apps</a> .

## Next steps

- Read more about [workload support](#).
- Get started with [Azure VM replication between regions](#).
- Get started with [VMware VM replication](#).

# Tutorial: Set up disaster recovery for Linux virtual machines

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This tutorial shows you how to set up disaster recovery for Azure VMs running Linux. In this article, learn how to:

- Enable disaster recovery for a Linux VM
- Run a disaster recovery drill to check it works as expected
- Stop replicating the VM after the drill

When you enable replication for a VM, the Site Recovery Mobility service extension installs on the VM, and registers it with [Azure Site Recovery](#). During replication, VM disk writes are sent to a cache storage account in the source VM region. Data is sent from there to the target region, and recovery points are generated from the data. When you fail a VM over to another region during disaster recovery, a recovery point is used to create a VM in the target region.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Prerequisites

1. Check that your Azure subscription allows you to create a VM in the target region. If you just created your free Azure account, you're the administrator of the subscription, and you have the permissions you need.
2. If you're not the subscription administrator, work with the administrator to assign you:
  - Either the Virtual Machine Contributor built-in role, or specific permissions to:
    - Create a VM in the selected virtual network.
    - Write to an Azure storage account.
    - Write to an Azure managed disk.
  - The Site Recovery Contributor built-in role, to manage Site Recovery operations in the vault.
3. Check that the Linux VM is running a [supported operating system](#).
4. If VM outbound connections use a URL-based proxy, make sure it can access these URLs. Using an authenticated proxy isn't supported.

Name	Public Cloud	Government Cloud	Details
Storage	*.blob.core.windows.net	*.blob.core.usgovcloudapi.net	Write data from the VM to the cache storage account in the source region.
Azure AD	login.microsoftonline.com	login.microsoftonline.us	Authorize and authenticate to Site Recovery service URLs.
Replication	*.hypervrecoverymanager.wi	*.hypervrecoverymanager.wi	Establish communication with the Site Recovery service.

Name	Public Cloud	Government Cloud	Details
Service Bus	*.servicebus.windows.net	*.servicebus.usgovcloudapi.net	Writes to Site Recovery for monitoring and diagnostic data.

5. If you're using network security groups (NSGs) to limit network traffic for VMs, create NSG rules that allow outbound connectivity (HTTPS 443) for the VM using these service tags (groups of IP addresses). Try out the rules on a test NSG first.

Tag	Allow
Storage tag	Allows data to be written from the VM to the cache storage account.
Azure AD tag	Allows access to all IP addresses that correspond to Azure AD.
EventsHub tag	Allows access to Site Recovery monitoring.
AzureSiteRecovery tag	Allows access to the Site Recovery service in any region.
GuestAndHybridManagement	Use if you want to automatically upgrade the Site Recovery Mobility agent that's running on VMs enabled for replication.

6. Make sure VMs have the latest root certificates. On Linux VMs, follow the guidance provided by your Linux distributor, to get the latest trusted root certificates and certificate revocation list on the VM.

## Create a VM and enable disaster recovery

You can optionally enable disaster recovery when you create a VM.

1. [Create a Linux VM](#).
2. On the **Management** tab, under **Site Recovery** select **Enable disaster recovery**.
3. In **Secondary region**, select the target region to which you want to replicate the VM for disaster recovery.
4. In **Secondary subscription**, select the target subscription in which the target VM will be created. The target VM is created when you fail over the source VM from the source region to the target region.
5. In **Recovery Services vault**, select the vault you want to use for the replication. If you don't have a vault, select **Create new**. Select a resource group in which to place the vault, and a vault name.
6. In **Site Recovery policy**, leave the default policy, or select **Create new** to set custom values.
  - Recovery points are created from snapshots of VM disks taken at a specific point in time. When you fail over a VM, you use a recovery point to restore the VM in the target region.
  - A crash-consistent recovery point is created every five minutes. This setting can't be modified. A crash-consistent snapshot captures data that was on the disk when the snapshot was taken. It doesn't include anything in memory.
  - By default Site Recovery keeps crash-consistent recovery points for 24 hours. You can set a custom value between 0 and 72 hours.
  - An app-consistent snapshot is taken every 4 hours.

- By default Site Recovery stores recovery points for 24 hours.
7. In **Availability options**, specify whether the VM is deployed as standalone, in an availability zone, or in an availability set.

The screenshot shows the 'Create a virtual machine' wizard in the Azure portal. The 'Site Recovery' section is highlighted, showing the following configuration:

- Enable Disaster Recovery:** Checked (indicated by a blue checkmark).
- Secondary region:** (Asia Pacific) South India
- Secondary subscription:** <subscription-name>
- Recovery Services vault:** ResourceMove-southeastasia-southindia-153c3e-0  
A red box highlights the 'Create new' button next to the vault name.
- Site Recovery policy:** RmsReplicationPolicy  
A red box highlights the 'Create new' button next to the policy name.
- Availability options:** No infrastructure redundancy required

A tooltip at the bottom left of the wizard window states: "By default, Azure Site Recovery will use the source machine's configuration for replication. After the virtual machine is created, you can edit these settings in 'Disaster recovery'. Click to learn more."

8. Finish creating the VM.

## Enable disaster recovery for an existing VM

If you want to enable disaster recovery on an existing VM, use this procedure.

1. In the Azure portal, open the VM properties page.
2. In **Operations**, select **Disaster recovery**.

Home > Virtual machines >

**azurevm2** Virtual machine

Search (Ctrl+ /) Connect Start Restart Stop Capture Delete Refresh

Diagnose and solve problems

**Settings**

- Networking
- Connect
- Disk
- Size
- Security
- Advisor recommendations
- Extensions
- Continuous delivery
- Availability + scaling
- Configuration
- Identity
- Properties
- Locks

**Operations**

- Bastion
- Auto-shutdown
- Backup

**Disaster recovery**

**Essentials**

Resource group (change) : <resource-group-name>  
Status : Running  
Location : West US  
Subscription (change) : <subscription-name>  
Subscription ID : <subscription-id>  
Tags (change) : Click here to add tags

**Properties** Monitoring Capabilities (7) Recommendations (10) Tutorials

**Virtual machine**

Computer name	azurevm2
Operating system	Windows (Windows Server 2012 R2 Datacenter)
Publisher	MicrosoftWindowsServer
Offer	WindowsServer
Plan	2012-R2-Datacenter
VM generation	V1
Agent status	Ready
Agent version	2.7.41491.1008
Host group	None
Host	-
Proximity placement group	-
Colocation status	N/A

3. In **Basics**, if the VM is deployed in an availability zone, you can select disaster recovery between availability zones.
4. In **Target region**, select the region to which you want to replicate the VM. The source and target regions must be in the same Azure Active Directory tenant.

Basics Advanced settings Review + Start replication



### Welcome to Azure Site Recovery

You can replicate your virtual machines to another Azure region for business continuity and disaster recovery needs. Replicate to the selected region so that you can recover your applications in the event of outages in source region. [Learn more about disaster recovery](#)

Disaster Recovery between Availability Zones? \* ⓘ

No

Target region \* ⓘ

West Europe



- 📍 Source region (North Europe)
- 📍 Selected target region (West Europe)
- 📍 Available target regions

[Review + Start replication](#)

Previous

Next : Advanced settings

5. Select **Next: Advanced settings**.

6. In **Advanced settings**, you can review settings, and modify values to custom settings. By default, Site Recovery mirrors the source settings to create target resources.

- **Target subscription.** The subscription in which the target VM is created after failover.
- **Target VM resource group.** The resource group in which the target VM is created after failover.
- **Target virtual network.** The Azure virtual network in which the target VM is located when it's created after failover.
- **Target availability.** When the target VM is created as a single instance, in an availability set, or availability zone.
- **Proximity placement.** If applicable, select the proximity placement group in which the target VM is located after failover.
- **Storage settings-Cache storage account.** Recovery uses a storage account in the source region as a temporary data store. Source VM changes are cached in this account, before being replicated to the target location.
  - By default one cache storage account is created per vault and reused.
  - You can select a different storage account if you want to customize the cache account for the VM.
- **Storage settings-Replica managed disk.** By default, Site Recovery creates replica managed disks

in the target region.

- By default the target managed disk mirror the source VM managed disks, using the same storage type (standard HDD/SSD, or premium SSD).
- You can customize the storage type as needed.
- **Replication settings.** Shows the vault in which the VM is located, and the replication policy used for the VM. By default, recovery points created by Site Recovery for the VM are kept for 24 hours.
- **Extension settings.** Indicates that Site Recovery manages updates to the Site Recovery Mobility Service extension that's installed on VMs you replicate.
  - The indicated Azure automation account manages the update process.
  - You can customize the automation account.

Basics   Advanced settings   Review + Start replication

**Target settings**

General settings	Source	Target	Info
Subscription	<subscription-name>	<subscription-name>	ⓘ
VM resource group	ABHISHEKRG-ASR	(new) ABHISHEKRG-ASR-asr	ⓘ
Virtual network	AbhishekRG-vnet	(new) AbhishekRG-vnet-asr	ⓘ
Availability	Availability zone	Single instance   Availability set Availability zone 3	ⓘ
Proximity placement	Not Applicable	Select	ⓘ

**Storage settings**   [-] Hide details

Cache storage account	(new) ezz1hzkeynoters5asrcache [Standard_LRS]	ⓘ		
Source managed disk	Replica managed disk	Replica managed disk	Disk to replicate	
[Premium SSD] Abhis...	(new) AbhishekVM_O...	Premium SSD	<input checked="" type="checkbox"/> include	ⓘ

**Replication settings**   [-] Hide details

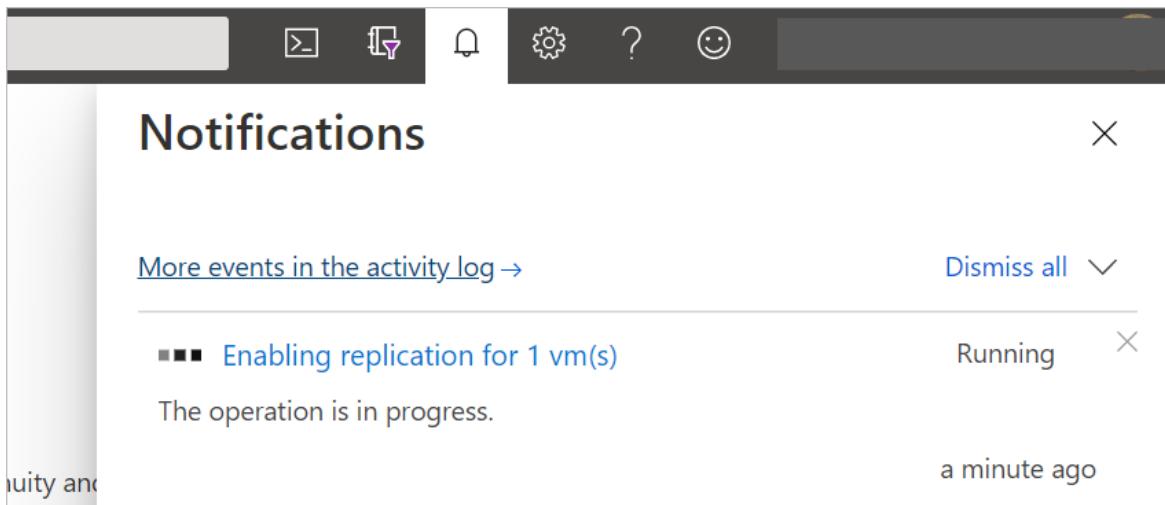
Vault subscription	<subscription-name>	ⓘ
Recovery services vault	<vault-name>	ⓘ
Vault resource group	KeyNoteMgmt	ⓘ
Replication policy	24-hour-retention-policy	ⓘ

**Extension settings**   [-] Hide details

Update settings	Allow ASR to manage	ⓘ
Automation account		ⓘ

7. Select **Review + Start replication**.

8. Select **Start replication**. Deployment starts, and Site Recovery starts creating target resources. You can monitor replication progress in the notifications.



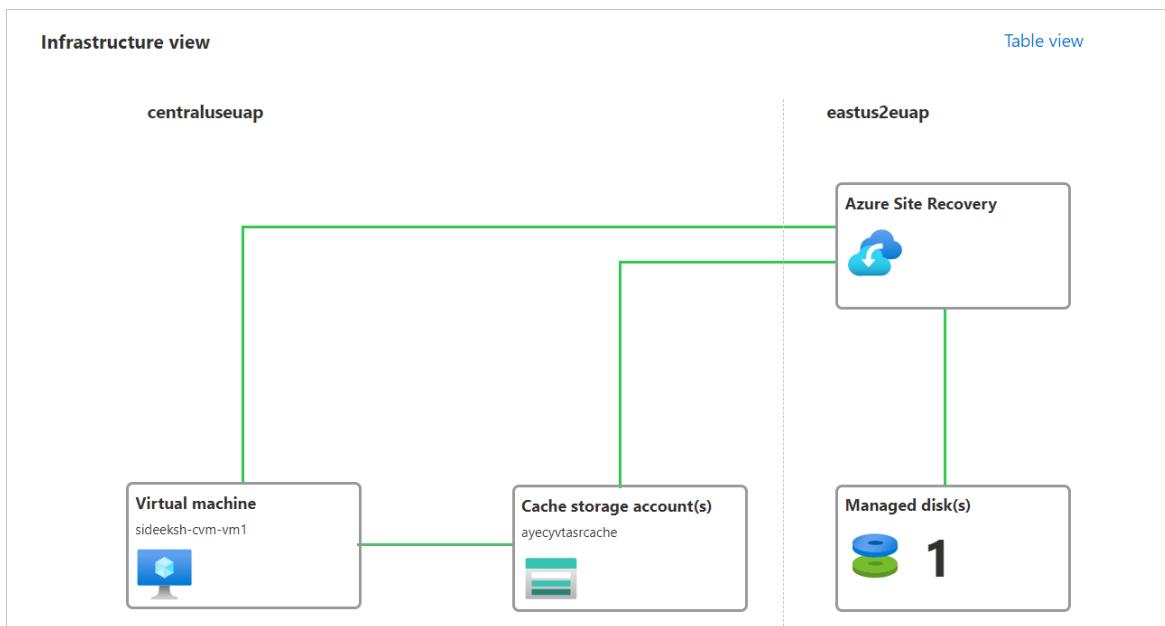
## Check VM status

After the replication job finishes, you can check the VM replication status.

1. Open the VM properties page.
2. In **Operations**, select **Disaster recovery**.
3. Expand the **Essentials** section to review defaults about the vault, replication policy, and target settings.
4. In **Health and status**, get information about replication state for the VM, the agent version, failover readiness, and the latest recovery points.

This screenshot shows the 'Health and status' section of the Azure portal. It includes tabs for Failover, Test Failover, Cleanup test failover, Commit, Resynchronize, Change recovery point, Re-protect, Disable Replication, Error Details, and Refresh. The 'Health and status' panel displays 'Replication Health' (Healthy, Protected, 1 min [As on 9/30/2020, 1:40:12 PM]), 'Failover readiness' (Last successful Test Failover, Configuration issues, Agent version, Agent status), and 'Latest recovery points' (Never performed successfully, No issues, 9.38.5739.1, Healthy). The 'Errors(0)' and 'Events - Last 72 hours(4)' sections show no errors or events.

5. In **Infrastructure view**, get a visual overview of source and target VMs, managed disks, and the cache storage account.



## Run a drill

Run a drill to make sure disaster recovery works as expected. When you run a test failover, it creates a copy of the VM, with no impact on ongoing replication, or on your production environment.

1. In the VM disaster recovery page, select **Test failover**.
2. In **Test failover**, leave the default **Latest processed (low RPO)** setting for the recovery point.

This option provides the lowest recovery point objective (RPO), and generally spins up the VM most quickly in the target region. It first processes all the data that has been sent to Site Recovery service, to create a recovery point for each VM, before failing over to it. This recovery point has all the data replicated to Site Recovery when the failover was triggered.

3. Select the virtual network in which the VM will be located after failover.

The screenshot shows the 'Test failover' configuration page for a VM named 'RayneTestVM-1'. The 'Failover direction' section shows 'From' set to 'East US 2' and 'To' set to 'West US'. Under 'Recovery Point', a dropdown menu is open, showing 'Latest processed (low RTO) (1 out of 1 disks) (10/1/2020, 10:53:40 AM)'. The 'Azure virtual network \*' section has a dropdown menu with 'Select' as the current choice. The URL at the top is 'Home > RayneTestVM-1 > RayneTestVM-1 > Test failover'.

4. The test failover process begins. You can monitor the progress in notifications.

The screenshot shows the 'Notifications' page with one active notification: 'Starting the test failover of 'RayneTestVM-1'...'. The status is 'Running' and the message says 'The operation is in progress.' with a timestamp 'a few seconds ago'. There is also a link 'More events in the activity log →' and a 'Dismiss all' button.

After the test failover completes, the VM is in the *Cleanup test failover pending* state on the Essentials

page.

## Clean up resources

The VM is automatically cleaned up by Site Recovery after the drill.

1. To begin automatic cleanup, select **Cleanup test failover**.

The screenshot shows the Site Recovery interface with the 'Cleanup test failover' button highlighted with a red box. The interface includes sections for Health and status, Failover readiness, and Latest recovery points. It also displays error and event logs.

2. In **Test failover cleanup**, type in any notes you want to record for the failover, and then select **Testing is complete. Delete test failover virtual machine**. Then select **OK**.

The dialog box shows the path Home > RayneTestVM-1 > Test failover cleanup. It contains a notes section with the text "Successful test failover. First try." and a checkbox labeled "Testing is complete. Delete test failover virtual machine(s.)".

3. The delete process begins. You can monitor progress in notifications.

# Notifications

X

[More events in the activity log →](#)

[Dismiss all](#)

■■■ Starting the task to delete the test failover environment... Running

The operation is in progress.

a few seconds ago

✓ Starting the test failover of 'RayneTestVM-1'...

Successfully completed the operation.

35 minutes ago

## Stop replicating the VM

After completing a disaster recovery drill, we suggest you continue to try out a full failover. If you don't want to do a full failover, you can disable replication. This does the following:

- Removes the VM from the Site Recovery list of replicated machines.
- Stops Site Recovery billing for the VM.
- Automatically cleans up source replication settings.

Stop replication as follows:

1. In the VM disaster recovery page, select **Disable Replication**.
2. In **Disable Replication**, select the reasons that you want to disable replication. Then select **OK**.

Home > RayneTestVM-1 >

### Disable Replication

RayneTestVM-1

This will remove the replicated item from Azure Site Recovery. Replication configuration on source will not be cleaned up. Site Recovery billing for the machine will stop. Click to learn more.

Please select the reason(s) for disabling protection for this virtual machine. Your feedback is important to improve our product to meet your requirements.

I don't want to provide feedback.

I completed migrating my application.

I am doing a proof of concept (POC) or trial with Azure Site Recovery.

Are you likely to use Azure Site Recovery in the future? \*

Yes

Please share feedback on what went well while using Azure Site Recovery and what did not?

I faced issues with Azure Site Recovery.

Other reasons

The Site Recovery extension installed on the VM during replication isn't removed automatically. If you disable replication for the VM, and you don't want to replicate it again at a later time, you can remove the Site Recovery extension manually, as follows:

1. Go to the VM > **Settings** > **Extensions**.
2. In the **Extensions** page, select each *Microsoft.Azure.RecoveryServices* entry for Linux.

3. In the properties page for the extension, select **Uninstall**.

## Next steps

In this tutorial, you configured disaster recovery for an Azure VM, and ran a disaster recovery drill. Now, you can perform a full failover for the VM.

[Fail over a VM to another region](#)

# Tutorial: Enable disaster recovery for Windows VMs

9/21/2022 • 8 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs ✓ Flexible scale sets

This tutorial shows you how to set up disaster recovery for Azure VMs running Windows. In this article, learn how to:

- Enable disaster recovery for a Windows VM
- Run a disaster recovery drill to check it works as expected
- Stop replicating the VM after the drill

When you enable replication for a VM, the Site Recovery Mobility service extension installs on the VM, and registers it with [Azure Site Recovery](#). During replication, VM disk writes are sent to a cache storage account in the source region. Data is sent from there to the target region, and recovery points are generated from the data. When you fail over a VM during disaster recovery, a recovery point is used to create a VM in the target region.

If you don't have an Azure subscription, create a [free account](#) before you begin.

## Prerequisites

1. Check that your Azure subscription allows you to create a VM in the target region. If you just created your free Azure account, you're the administrator of the subscription, and you have the permissions you need.
2. If you're not the subscription administrator, work with the administrator to assign you:
  - Either the Virtual Machine Contributor built-in role, or specific permissions to:
    - Create a VM in the selected virtual network.
    - Write to an Azure storage account.
    - Write to an Azure-managed disk.
  - The Site Recovery Contributor built-in role, to manage Site Recovery operations in the vault.
3. We recommend you use a Windows VM running Windows Server 2012 or later. The VM disk shouldn't be encrypted for the purpose of this tutorial.
4. If VM outbound connections use a URL-based proxy, make sure it can access these URLs. Using an authenticated proxy isn't supported.

NAME	PUBLIC CLOUD	GOVERNMENT CLOUD	DETAILS
Storage	*.blob.core.windows.net	*.blob.core.usgovcloudapi.net	Write data from the VM to the cache storage account in the source region.
Azure AD	login.microsoftonline.com	login.microsoftonline.us	Authorize and authenticate to Site Recovery service URLs.
Replication	*.hypervrecoverymanager.wi	*.hypervrecoverymanager.wi	VM communication with the Site Recovery service.

Name	Public Cloud	Government Cloud	Details
Service Bus	*.servicebus.windows.net	*.servicebus.usgovcloudapi. <sup>VM</sup>	writes to Site Recovery monitoring and diagnostic data.

5. If you're using network security groups (NSGs) to limit network traffic for VMs, create NSG rules that allow outbound connectivity (HTTPS 443) for the VM using these service tags (groups of IP addresses). Try out the rules on a test NSG first.

Tag	Allow
Storage tag	Allows data to be written from the VM to the cache storage account.
Azure AD tag	Allows access to all IP addresses that correspond to Azure AD.
EventsHub tag	Allows access to Site Recovery monitoring.
AzureSiteRecovery tag	Allows access to the Site Recovery service in any region.
GuestAndHybridManagement	Use if you want to automatically upgrade the Site Recovery Mobility agent that's running on VMs enabled for replication.

6. On Windows VMs, install the latest Windows updates, to make sure that VMs have the latest root certificates.

## Create a VM and enable disaster recovery

You can optionally enable disaster recovery when you create a VM.

1. [Create a VM](#).
2. On the **Management** tab, select **Enable disaster recovery**.
3. In **Secondary region**, select the target region to which you want to replicate a VM for disaster recovery.
4. In **Secondary subscription**, select the target subscription in which the target VM will be created. The target VM is created when you fail over the source VM from the source region to the target region.
5. In **Recovery Services vault**, select the vault you want to use for the replication. If you don't have a vault, select **Create new**. Select a resource group in which to place the vault, and a vault name.
6. In **Site Recovery policy**, leave the default policy, or select **Create new** to set custom values.
  - Recovery points are created from snapshots of VM disks taken at a specific point in time. When you fail over a VM, you use a recovery point to restore the VM in the target region.
  - A crash-consistent recovery point is created every five minutes. This setting can't be modified. A crash-consistent snapshot captures data that was on the disk when the snapshot was taken. It doesn't include anything in memory.
  - By default Site Recovery keeps crash-consistent recovery points for 24 hours. You can set a custom value between 0 and 72 hours.
  - An app-consistent snapshot is taken every 4 hours. An app-consistent snapshot
  - By default Site Recovery stores recovery points for 24 hours.

7. In **Availability options**, specify whether the VM is deployed as standalone, in an availability zone, or in an availability set.

The screenshot shows the 'Create a virtual machine' wizard in the Azure portal. The 'Site Recovery' section is highlighted, indicating it's the current step. Under 'Site Recovery', the 'Enable Disaster Recovery' checkbox is checked. Below it, the 'Secondary region' dropdown is set to '(Asia Pacific) South India'. The 'Secondary subscription' dropdown shows '<subscription-name>' and the 'Recovery Services vault' dropdown shows 'ResourceMove-southeastasia-southindia-153c3e-0'. Both the 'Site Recovery policy' and 'Availability options' dropdowns show 'Create new' and 'No infrastructure redundancy required' respectively. A note at the bottom states: 'By default, Azure Site Recovery will use the source machine's configuration for replication. After the virtual machine is created, you can edit these settings in "Disaster recovery". Click to learn more.'

8. Finish creating the VM.

#### NOTE

When you enable replication while creating a Windows VM, only the OS disk gets replicated. Data disks need to be initialized by you, after which Azure Site Recovery automatically replicates them.

## Enable disaster recovery for an existing VM

If you want to enable disaster recovery on an existing VM instead of for a new VM, use this procedure.

1. In the Azure portal, open the VM properties page.
2. In **Operations**, select **Disaster recovery**.

Home > Virtual machines >

**azurevm2** Virtual machine

Search (Ctrl+ /) Connect Start Restart Stop Capture Delete Refresh

Diagnose and solve problems

**Settings**

- Networking
- Connect
- Disk
- Size
- Security
- Advisor recommendations
- Extensions
- Continuous delivery
- Availability + scaling
- Configuration
- Identity
- Properties
- Locks

**Operations**

- Bastion
- Auto-shutdown
- Backup

**Disaster recovery**

**Essentials**

Resource group (change) : <resource-group-name>  
Status : Running  
Location : West US  
Subscription (change) : <subscription-name>  
Subscription ID : <subscription-id>  
Tags (change) : Click here to add tags

**Properties** Monitoring Capabilities (7) Recommendations (10) Tutorials

**Virtual machine**

Computer name	azurevm2
Operating system	Windows (Windows Server 2012 R2 Datacenter)
Publisher	MicrosoftWindowsServer
Offer	WindowsServer
Plan	2012-R2-Datacenter
VM generation	V1
Agent status	Ready
Agent version	2.7.41491.1008
Host group	None
Host	-
Proximity placement group	-
Colocation status	N/A

3. In **Basics**, if the VM is deployed in an availability zone, you can select disaster recovery between availability zones.
4. In **Target region**, select the region to which you want to replicate the VM. The source and target regions must be in the same Azure Active Directory tenant.

Basics Advanced settings Review + Start replication



### Welcome to Azure Site Recovery

You can replicate your virtual machines to another Azure region for business continuity and disaster recovery needs. Replicate to the selected region so that you can recover your applications in the event of outages in source region. [Learn more about Site Recovery](#)

Disaster Recovery between Availability Zones? \* ⓘ

No

Target region \* ⓘ

West Europe



- 📍 Source region (North Europe)
- 📍 Selected target region (West Europe)
- 📍 Available target regions

[Review + Start replication](#)

[Previous](#)

[Next : Advanced settings](#)

5. Select **Next: Advanced settings**.

6. In **Advanced settings**, you can review settings, and modify values to custom settings. By default, Site Recovery mirrors the source settings to create target resources.

- **Target subscription.** The subscription in which the target VM is created after failover.
- **Target VM resource group.** The resource group in which the target VM is created after failover.
- **Target virtual network.** The Azure virtual network in which the target VM is located when it's created after failover.
- **Target availability.** When the target VM is created as a single instance, in an availability set, or availability zone.
- **Proximity placement.** If applicable, select the proximity placement group in which the target VM is located after failover.
- **Storage settings-Cache storage account.** Recovery uses a storage account in the source region as a temporary data store. Source VM changes are cached in this account, before being replicated to the target location.
  - By default one cache storage account is created per vault and reused.
  - You can select a different storage account if you want to customize the cache account for the VM.
- **Storage settings-Replica managed disk.** By default, Site Recovery creates replica managed disks

in the target region.

- By default the target managed disk mirror the source VM managed disks, using the same storage type (standard HDD/SSD, or premium SSD).
- You can customize the storage type as needed.
- **Replication settings.** Shows the vault in which the VM is located, and the replication policy used for the VM. By default, recovery points created by Site Recovery for the VM are kept for 24 hours.
- **Extension settings.** Indicates that Site Recovery manages updates to the Site Recovery Mobility Service extension that's installed on VMs you replicate.
  - The indicated Azure automation account manages the update process.
  - You can customize the automation account.

Basics   Advanced settings   Review + Start replication

**Target settings**

General settings	Source	Target	Info
Subscription	<subscription-name>	<subscription-name>	(i)
VM resource group	ABHISHEKRG-ASR	(new) ABHISHEKRG-ASR-asr	(i)
Virtual network	AbhishekRG-vnet	(new) AbhishekRG-vnet-asr	(i)
Availability	Availability zone 3	Single instance   Availability set Availability zone 3	(i)
Proximity placement	Not Applicable	Select	(i)

**Storage settings**   [-] Hide details

Cache storage account	(new) ezz1hzkeynoters5asrcache [Standard_LRS]	(i)		
Source managed disk	Replica managed disk	Replica managed disk	Disk to replicate	
[Premium SSD] Abhis...	(new) AbhishekVM_O...	Premium SSD	<input checked="" type="checkbox"/> include	(i)

**Replication settings**   [-] Hide details

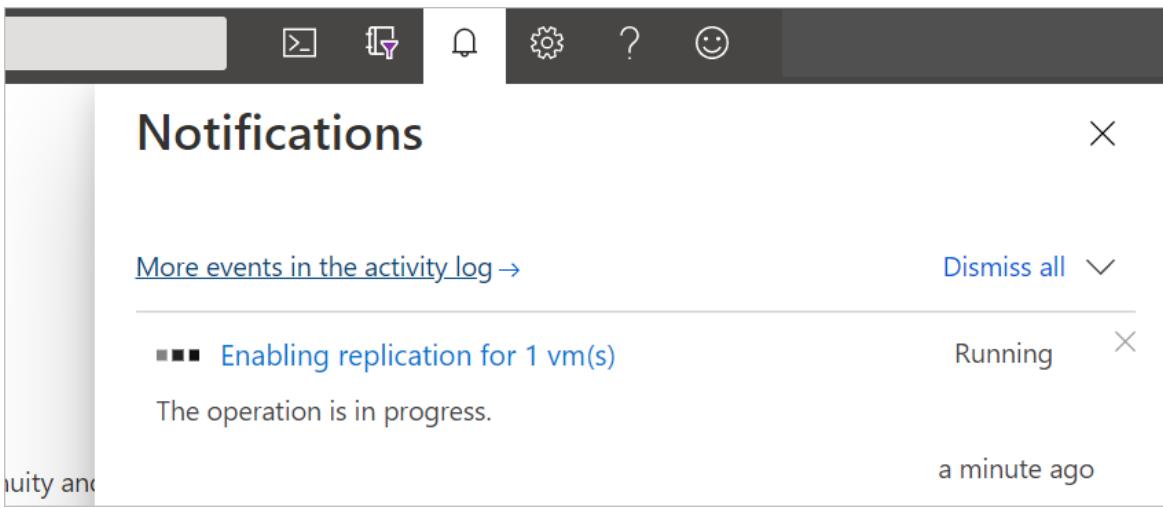
Vault subscription	<subscription-name>	(i)
Recovery services vault	<vault-name>	(i)
Vault resource group	KeyNoteMgmt	(i)
Replication policy	24-hour-retention-policy	(i)

**Extension settings**   [-] Hide details

Update settings	Allow ASR to manage	(i)
Automation account		(i)

7. Select **Review + Start replication**.

8. Select **Start replication**. Deployment starts, and Site Recovery starts creating target resources. You can monitor replication progress in the notifications.



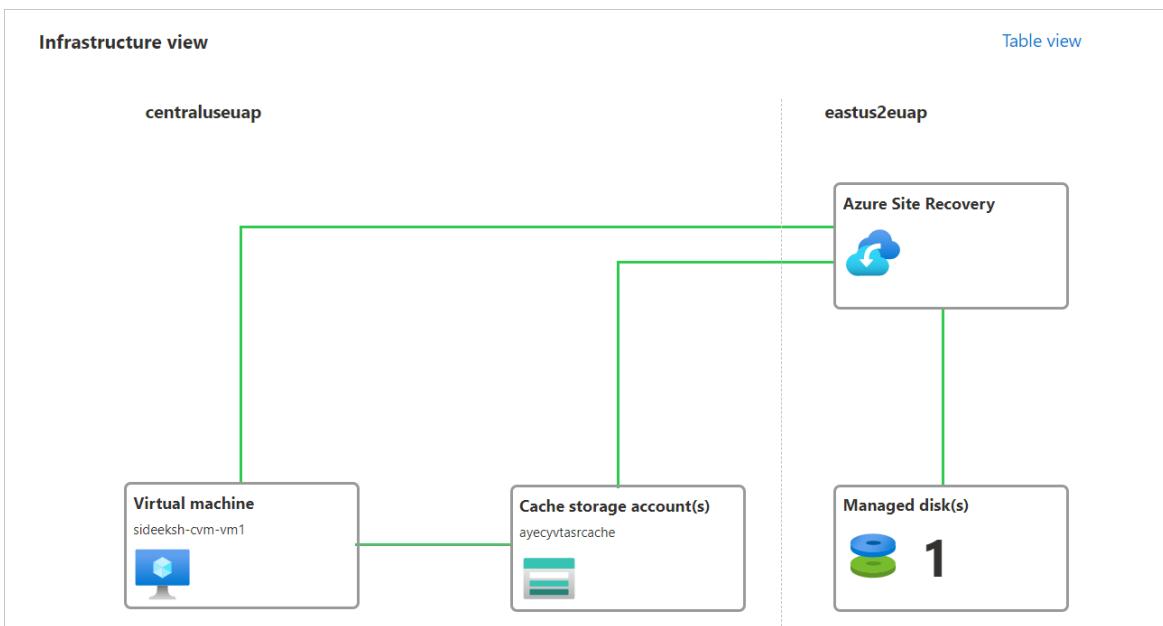
## Check VM status

After the replication job finishes, you can check the VM replication status.

1. Open the VM properties page.
2. In **Operations**, select **Disaster recovery**.
3. Expand the **Essentials** section to review defaults about the vault, replication policy, and target settings.
4. In **Health and status**, get information about replication state for the VM, the agent version, failover readiness, and the latest recovery points.

This screenshot shows the 'Health and status' blade for disaster recovery. It includes tabs for Failover, Test Failover, Cleanup test failover, Commit, Resynchronize, Change recovery point, Re-protect, Disable Replication, Error Details, and Refresh. The main area has sections for 'Health and status' (Replication Health: Healthy, Status: Protected, RPO: 1 min [As on 9/30/2020, 1:40:12 PM]), 'Failover readiness' (Last successful Test Failover: Never performed successfully, Configuration issues: No issues, Agent version: 9.38.5739.1, Agent status: Healthy), and 'Latest recovery points' (Click above to see the latest recovery points). At the bottom, there are sections for Errors (0) and Events - Last 72 hours (4).

5. In **Infrastructure view**, get a visual overview of source and target VMs, managed disks, and the cache storage account.



## Run a drill

Run a drill to make sure disaster recovery works as expected. When you run a test failover, it creates a copy of the VM, with no impact on ongoing replication, or on your production environment.

1. In the VM disaster recovery page, select **Test failover**.
2. In **Test failover**, leave the default **Latest processed (low RPO)** setting for the recovery point.

This option provides the lowest recovery point objective (RPO), and generally the quickest spin up of the target VM. It first processes all the data that has been sent to Site Recovery service, to create a recovery point for each VM, before failing over to it. This recovery point has all the data replicated to Site Recovery when the failover was triggered.

3. Select the virtual network in which the VM will be located after failover.

The screenshot shows the 'Test failover' configuration page for a VM named 'RayneTestVM-1'. The 'Failover direction' section shows 'From' set to 'East US 2' and 'To' set to 'West US'. Under 'Recovery Point', a dropdown menu is open, showing 'Latest processed (low RTO) (1 out of 1 disks) (10/1/2020, 10:53:40 AM)'. The 'Azure virtual network \*' section has a dropdown menu labeled 'Select'.

4. The test failover process begins. You can monitor the progress in notifications.

The screenshot shows the 'Notifications' window. It displays a single notification: 'Starting the test failover of 'RayneTestVM-1'...' with a status of 'Running'. Below the notification, it says 'The operation is in progress.' and 'a few seconds ago'. There is also a link to 'More events in the activity log' and a 'Dismiss all' button.

After the test failover completes, the VM is in the *Cleanup test failover pending* state on the **Essentials**

page.

## Clean up resources

The VM is automatically cleaned up by Site Recovery after the drill.

1. To begin automatic cleanup, select **Cleanup test failover**.

The screenshot shows the Site Recovery interface with the 'Cleanup test failover' button highlighted with a red box. The interface includes sections for Health and status, Failover readiness, and Latest recovery points. It also displays error and event logs.

2. In **Test failover cleanup**, type in any notes you want to record for the failover, and then select **Testing is complete. Delete test failover virtual machine**. Then select **OK**.

The dialog box shows the path Home > RayneTestVM-1 > Test failover cleanup. It contains a notes section with the text "Successful test failover. First try." and a checkbox labeled "Testing is complete. Delete test failover virtual machine(s.)".

3. The delete process begins. You can monitor progress in notifications.

# Notifications



[More events in the activity log →](#)

[Dismiss all](#)

■■■ Starting the task to delete the test failover environment... Running

The operation is in progress.

a few seconds ago

✓ Starting the test failover of 'RayneTestVM-1'...

Successfully completed the operation.

35 minutes ago

## Stop replicating the VM

After completing a disaster recovery drill, we suggest you continue to try out a full failover. If you don't want to do a full failover, you can disable replication. This does the following:

- Removes the VM from the Site Recovery list of replicated machines.
- Stops Site Recovery billing for the VM.
- Automatically cleans up source replication settings.

Stop replication as follows:

1. In the VM disaster recovery page, select **Disable Replication**.
2. In **Disable Replication**, select the reasons that you want to disable replication. Then select **OK**.

Home > RayneTestVM-1 >

### Disable Replication

RayneTestVM-1

This will remove the replicated item from Azure Site Recovery. Replication configuration on source will not be cleaned up. Site Recovery billing for the machine will stop. Click to learn more.

Please select the reason(s) for disabling protection for this virtual machine. Your feedback is important to improve our product to meet your requirements.

I don't want to provide feedback.

I completed migrating my application.

I am doing a proof of concept (POC) or trial with Azure Site Recovery.

Are you likely to use Azure Site Recovery in the future? \*

Yes

Please share feedback on what went well while using Azure Site Recovery and what did not?

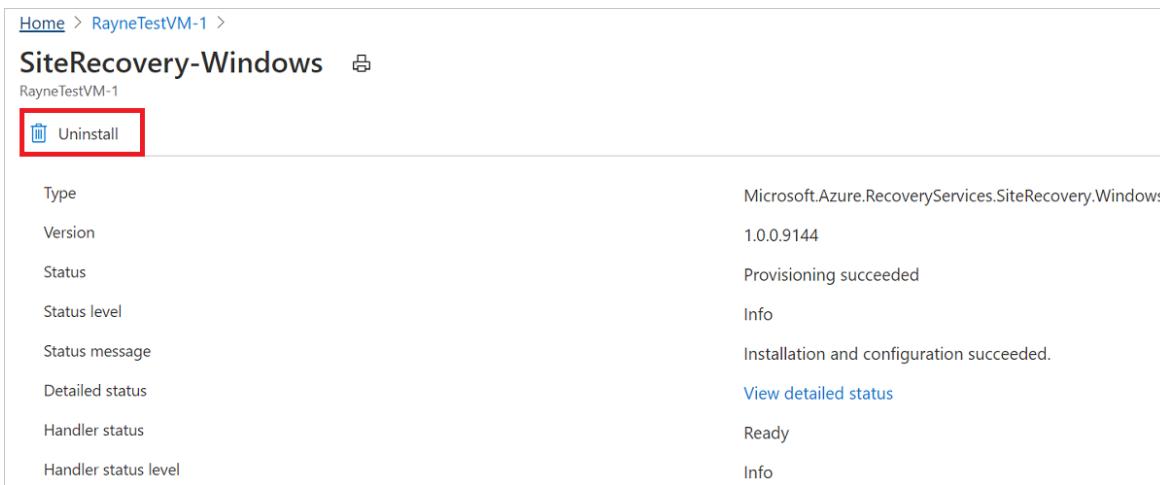
I faced issues with Azure Site Recovery.

Other reasons

The Site Recovery extension installed on the VM during replication isn't removed automatically. If you disable replication for the VM, and you don't want to replicate it again at a later time, you can remove the Site Recovery extension manually, as follows:

1. Go to the VM > **Settings** > **Extensions**.
2. In the **Extensions** page, select each *Microsoft.Azure.RecoveryServices* entry for Linux.

3. In the properties page for the extension, select **Uninstall**.



The screenshot shows the Azure portal interface for managing extensions. At the top, there's a breadcrumb navigation: Home > RayneTestVM-1 >. Below it, the title is SiteRecovery-Windows. Underneath the title, it says RayneTestVM-1. A prominent red box highlights the "Uninstall" button, which has a trash can icon. To the right of the button is a table with the following data:

Type	Microsoft.Azure.RecoveryServices.SiteRecovery.Windows
Version	1.0.0.9144
Status	Provisioning succeeded
Status level	Info
Status message	Installation and configuration succeeded.
Detailed status	<a href="#">View detailed status</a>
Handler status	Ready
Handler status level	Info

## Next steps

In this tutorial, you configured disaster recovery for an Azure VM, and ran a disaster recovery drill. Now, you can perform a full failover for the VM.

[Fail over a VM to another region](#)

# Tutorial: Fail over Azure VMs to a secondary region

9/21/2022 • 3 minutes to read • [Edit Online](#)

Learn how to fail over Azure VMs that are enabled for disaster recovery with [Azure Site Recovery](#), to a secondary Azure region. After failover, you reprotect VMs in the target region so that they replicate back to the primary region. In this article, you learn how to:

- Check prerequisites
- Verify VM settings
- Run a failover to the secondary region
- Start replicating the VM back to the primary region.

## NOTE

This tutorial shows you how to fail over VMs with minimal steps. If you want to run a failover with full settings, learn about Azure VM [networking](#), [automation](#), and [troubleshooting](#).

## Prerequisites

Before you start this tutorial, you should have:

1. Set up replication for one or more Azure VMs. If you haven't, [complete the first tutorial](#) in this series to do that.
2. We recommend you [run a disaster recovery drill](#) for replicated VMs. Running a drill before you run a full failover helps ensure everything works as expected, without impacting your production environment.

## Verify the VM settings

1. In the vault > **Replicated items**, select the VM.

Name	Replication Health	Status	Active location
RayneTest-VM-EastUS	Healthy	Protected	East US

2. On the **VM Overview** page, check that the VM is protected and healthy, before you run a failover.

3. Before you fail over, check that:

- The VM is running a supported [Windows](#) or [Linux](#) operating system.
- The VM complies with [compute](#), [storage](#), and [networking](#) requirements.

# Run a failover

1. On the VM Overview page, select Failover.

The screenshot shows the Azure VM Overview page for 'AdminVM1'. The 'Failover' tab is selected in the top navigation bar. In the main content area, there's a 'Health and status' summary table and a 'Failover readiness' table. The 'Failover readiness' table includes columns for 'Last successful Test Failover' (Never performed successfully), 'Configuration issues' (No issues), 'Agent version' (9.45.6096.1), and 'Agent status' (Healthy).

Health and status		Failover readiness	
Replication Health	Healthy	Last successful Test Failover	Never performed successfully
Status	Protected	Configuration issues	No issues
RPO	2 mins [As on 10/1/2021, 5:56:33 PM]	Agent version	9.45.6096.1
		Agent status	Healthy

2. In Failover, choose a recovery point. The Azure VM in the target region is created using data from this recovery point.

- **Latest processed:** Uses the latest recovery point processed by Site Recovery. The time stamp is shown. No time is spent processing data, so it provides a low recovery time objective (RTO).
- **Latest:** Processes all the data sent to Site Recovery, to create a recovery point for each VM before failing over to it. Provides the lowest recovery point objective (RPO), because all data is replicated to Site Recovery when the failover is triggered.
- **Latest app-consistent:** This option fails over VMs to the latest app-consistent recovery point. The time stamp is shown.
- **Custom:** Fail over to particular recovery point. Custom is only available when you fail over a single VM, and don't use a recovery plan.

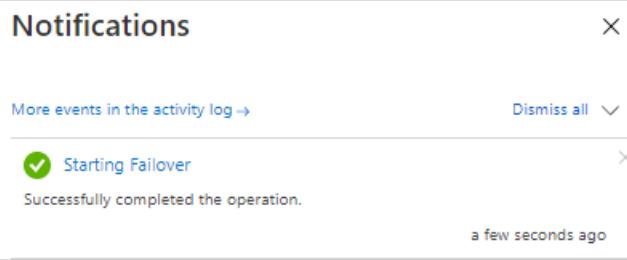
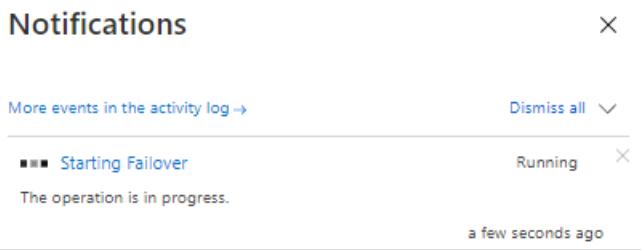
## NOTE

If you added a disk to a VM after you enabled replication, replication points shows disks available for recovery. For example, a replication point created before you added a second disk will show as "1 of 2 disks".

3. Select **Shut down machine before beginning failover** if you want Site Recovery to try to shut down the source VMs before starting failover. Shutdown helps to ensure no data loss. Failover continues even if shutdown fails.

The screenshot shows the 'Failover' configuration dialog for 'RayneTest-VMEastUS'. It includes fields for 'Failover direction' (From: East US(Zone null), To: East US 2(Zone null)), 'Recovery Point' (Choose a recovery point: Latest processed (low RTO) (1 out of 1 ...)), and a checked checkbox for 'Shut down machine before beginning failover.'

4. To start the failover, select OK.
5. Monitor the failover in notifications.



6. After the failover, the Azure VM created in the target region appears in **Virtual Machines**. Make sure that the VM is running, and sized appropriately. If you want to use a different recovery point for the VM, select **Change recovery point**, on the **Essentials** page.
7. When you're satisfied with the failed over VM, select **Commit** on the overview page, to finish the failover.

The screenshot shows the 'Site Recovery' blade in the Azure portal. At the top, there are several buttons: 'Failover', 'Test Failover', 'Cleanup test failover', 'Commit' (which is highlighted with a red box), 'Resynchronize', 'Change recovery point', 'Re-protect', and 'Disable Replication'. Below this, there's a section titled 'Essentials' with two main sections: 'Health and status' and 'Failover readiness'. Under 'Health and status', it shows 'Replication Health' as '-' and 'Status' as 'Failover completed'. Under 'Failover readiness', it shows 'Last successful Test Failover', 'Configuration issues', 'Agent version', and 'Agent status'.

8. In **Commit**, select **OK** to confirm. Commit deletes all the available recovery points for the VM in Site Recovery, and you won't be able to change the recovery point.
9. Monitor the commit progress in notifications.

The first screenshot shows a 'NOTIFICATIONS' window with a single entry: 'Committing Failover' status 'Running'. The message says 'The operation is in progress.' and was 'a few seconds ago'. The second screenshot shows a 'Notifications' window with a single entry: 'Committing Failover' status 'Successfully completed the operation.' and was 'a few seconds ago'.

## Reprotect the VM

After failover, you reprotect the VM in the secondary region, so that it replicates back to the primary region.

1. Make sure that **VM Status** is *Failover committed* before you start.

2. Check that you can access the primary region is available, and that you have permissions to create VMs in it.

3. On the VM Overview page, select Re-Protect.

The screenshot shows the 'RayneTest-VMEastUS' VM Overview page in the Azure portal. The 'Re-protect' button is highlighted with a red box. Other buttons visible include Failover, Test Failover, Cleanup test failover, Commit, Resynchronize, Change recovery point, Disable Replication, Error Details, and Refresh.

4. In Re-protect, verify the replication direction (secondary to primary region), and review the target settings for the primary region. Resources marked as new are created by Site Recovery as part of the reprotect operation.

The screenshot shows the 'Re-protect' configuration dialog. It displays the replication direction as 'eastus to eastus2'. A warning message states: '⚠ If you are choosing General Purpose v2 storage accounts, ensure that operations and data transfer prices are understood clearly before you proceed. [Learn more](#)'. Below this, there are sections for 'Resource group, Network, Storage and Availability' (Customize) and 'Target resource group' (RayneTest-EastUSRG). The 'Cache storage accounts' section lists 'qikc7eraynetestvasrcache'. The 'Target virtual network' is set to 'RayneTest-EastUSRG-vnet'. Under 'Replica managed disks', it shows '(new) 1 premium disk(s), 0 standard disk(s)'. The 'Target availability sets' section indicates 'Not Applicable'.

5. Select OK to start the reprotect process. The process sends initial data to the target location, and then replicates delta information for the VMs to the target.

6. Monitor reprotect progress in the notifications.

The first screenshot shows a 'Notifications' pane with one event: 'Reprotecting virtual machine' (Running, 'The operation is in progress.' a few seconds ago). The second screenshot shows the same pane after completion, with the event status changed to 'Succeeded' (Successfully completed the operation. 8 minutes ago).

## Next steps

In this tutorial, you failed over from the primary region to the secondary, and started replicating VMs back to the primary region. Now you can fail back from the secondary region to the primary.

[Fail back to the primary region](#)

# Tutorial: Fail back Azure VM to the primary region

9/21/2022 • 3 minutes to read • [Edit Online](#)

After failing over an Azure VM to a secondary Azure region, follow this tutorial to fail the VM to the primary Azure region, using [Azure Site Recovery](#). In this article, you learn how to:

- Review the prerequisites.
- Fail back the VM in the secondary region.
- Reprotect primary VMs back to the secondary region.

## NOTE

This tutorial shows you how to fail back with minimal steps. If you want to run a failover with full settings, learn about Azure VM [networking](#), [automation](#), and [troubleshooting](#).

## Prerequisites

Before you start this tutorial, you should have:

1. [Set up replication](#) for at least one Azure VM, and tried out a [disaster recovery drill](#) for it.
2. [Failed over the VM](#) from the primary region to a secondary region, and reprotected it so that it replicates from the secondary region to the primary.
3. Check that the primary region is available, and that you're able to create and access new resources in it.

## Fail back to the primary region

After VMs are reprotected, you can fail back to the primary region as needed.

1. In the vault > **Replicated items**, select the VM.
2. On the VM overview page, check that the VM is healthy, and that synchronization is complete, before you run a failover. The VM should be in a *Protected* state.

The screenshot shows the Azure portal interface for a replicated item named "AdminVM1". The "Overview" tab is active. Key details shown include:

- Health and status:** Replication Health is "Healthy", Status is "Protected", RPO is "1 min [As on 10/1/2021, 7:35:56 PM]".
- Failover readiness:** Last successful Test Failover is "Never performed successfully". Configuration issues, Agent version (9.45.6096.1), and Agent status are all marked as "No issues".
- Action bar:** Includes buttons for Failover, Test Failover, Cleanup test failover, Commit, Resynchronize, Change recovery point, Re-protect, and a refresh icon.

3. On the overview page, select **Failover**. Since we're not doing a test failover this time, we're prompted to verify.

[Page showing we agree to run failover without a test failover](#)

4. In **Failover**, note the direction from secondary to primary, and select a recovery point. The Azure VM in the target (primary region) is created using data from this point.
  - **Latest processed:** Uses the latest recovery point processed by Site Recovery. The time stamp is

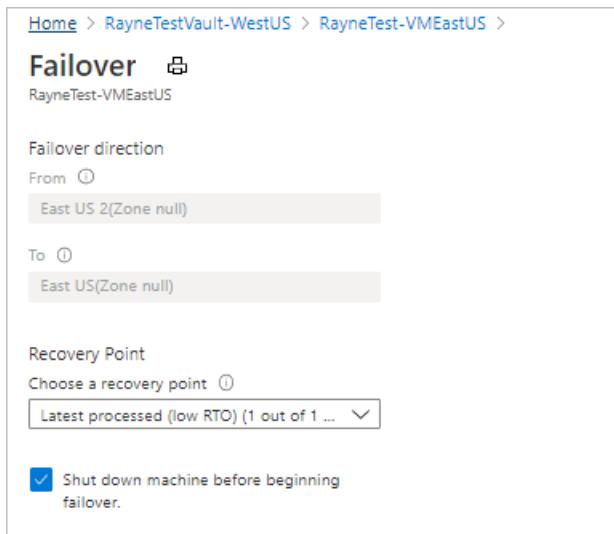
shown. No time is spent processing data, so it provides a low recovery time objective (RTO).

- **Latest**: Processes all the data sent to Site Recovery, to create a recovery point for each VM before failing over to it. Provides the lowest recovery point objective (RPO), because all data is replicated to Site Recovery when the failover is triggered.
- **Latest app-consistent**: This option fails over VMs to the latest app-consistent recovery point. The time stamp is shown.
- **Custom**: Fail over to particular recovery point. Custom is only available when you fail over a single VM, and don't use a recovery plan.

#### NOTE

If you fail over a VM to which you added a disk after you enabled replication for the VM, replication points will show the disks available for recovery. For example, a replication point that was created before you added a second disk will show as "1 of 2 disks".

5. Select **Shut down machine before beginning failover** if you want Site Recovery to attempt to shut down the source VMs before starting failover. Shutdown helps to ensure no data loss. Failover continues even if shutdown fails.



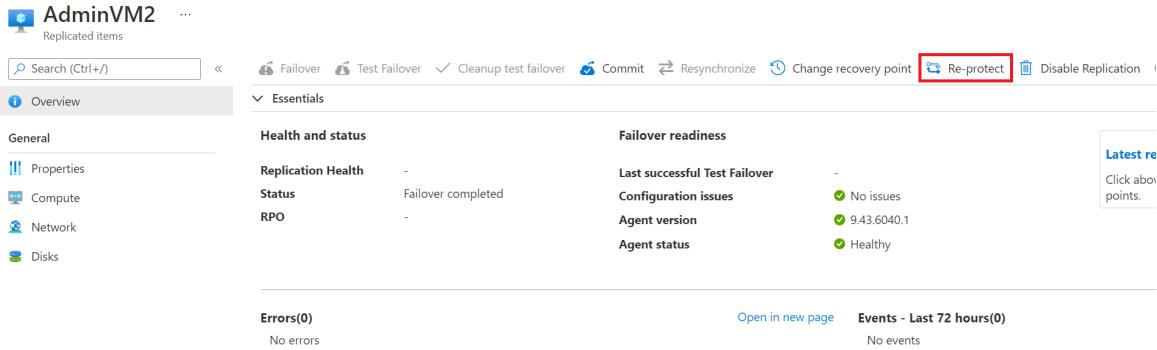
6. To start the failover, select OK.
7. Monitor the failover in notifications.

The first screenshot shows a 'Notifications' dialog box with one item: 'Starting Failover' (Running, 'The operation is in progress.' a few seconds ago). The second screenshot shows the same dialog box with the same item, but with a checkmark icon and the message 'Successfully completed the operation.'

# Reprotect VMs

After failing back VMs to the primary region, you need to reprotect them, so that they start replicating to the secondary region again.

1. In the **Overview** page for the VM, select **Re-protect**.



The screenshot shows the 'Overview' page for a replicated item named 'AdminVM2'. The top navigation bar includes buttons for Failover, Test Failover, Cleanup test failover, Commit, Resynchronize, Change recovery point, Re-protect (which is highlighted with a red box), and Disable Replication. The main content area is divided into sections: General, Health and status, Failover readiness, Errors(0), and Events - Last 72 hours(0). The 'Health and status' section shows Replication Health as 'Completed', Status as 'Failover completed', and RPO as '-'. The 'Failover readiness' section shows 'Last successful Test Failover' as '-' and 'Configuration issues' as 'No issues'. The 'Errors(0)' section indicates 'No errors' with a link to 'Open in new page'. The 'Events - Last 72 hours(0)' section indicates 'No events'.

2. Review the target settings for the primary region. Resources marked as new are created by Site Recovery as part of the reprotect operation.
3. Select **OK** to start the reprotect process. The process sends initial data to the target location, and then replicates delta information for the VMs to the target.

## Re-protect



eastus to eastus2



**!** If you are choosing General Purpose v2 storage accounts, ensure that operations and data transfer prices are understood clearly before you proceed. [Learn more](#)

Resource group, Network, Storage and Availability [Customize](#)

By default, Site Recovery will pick the original source resource group, virtual network, storage accounts and availability sets as below. Click 'Customize' above to change the configuration. The resources created are appended with "asr" suffix.

**Target resource group** ⓘ

RayneTest-EastUSRG-asr-1

**Target virtual network** ⓘ

RayneTest-EastUSRG-vnet-asr

**Cache storage accounts** ⓘ

qikc7eraynetestvasrcache

**Replica managed disks** ⓘ

(new) 1 premium disk(s), 0 standard disk(s)

**Target availability sets** ⓘ

Not Applicable

**OK**

4. Monitor reprotect progress in notifications.

## Notifications

[More events in the activity log →](#)

[Dismiss all](#)

■■■ [Reprotecting virtual machine](#)

Running

The operation is in progress.

[Reprotect progress](#)

notification

## Clean up resources

For VMs with managed disks, after failback is complete and VMs are reprotected for replication from primary to secondary, Site Recovery automatically cleans up machines in the secondary disaster recovery region. You don't need to manually delete VMs and NICs in the secondary region. VMs with unmanaged disks aren't cleaned up.

If you completely disable replication after failing back, Site Recovery cleans up machines protected by it. In this case, it also cleans up disks for VMs that don't use managed disks.

## Next steps

In this tutorial, you failed VMs back from the secondary region to the primary. This is the last step in the process that includes enabling replication for a VM, trying out a disaster recovery drill, failing over from the primary region to the secondary, and finally failing back.

Now, try out disaster recovery to Azure for an [on-premises VM](#)

# Overview of VM restore points

9/21/2022 • 3 minutes to read • [Edit Online](#)

Business continuity and disaster recovery (BCDR) solutions are primarily designed to address site-wide data loss. Solutions that operate at this scale will often manage and execute automated failovers and failbacks across multiple regions. Azure VM restore points can be used to implement granular backup and retention policies.

You can protect your data and guard against extended downtime by creating virtual machine (VM) restore points at regular intervals. There are several backup options available for virtual machines (VMs), depending on your use-case. For more information, see [Backup and restore options for virtual machines in Azure](#).

## About VM restore points

An individual VM restore point is a resource that stores VM configuration and point-in-time application consistent snapshots of all the managed disks attached to the VM. You can use VM restore points to easily capture multi-disk consistent backups. VM restore points contain a disk restore point for each of the attached disks and a disk restore point consists of a snapshot of an individual managed disk.

VM restore points support application consistency for VMs running Windows operating systems and support file system consistency for VMs running Linux operating system. Application consistent restore points use VSS writers (or pre/post scripts for Linux) to ensure the consistency of the application data before a restore point is created. To get an application consistent restore point, the application running in the VM needs to provide a VSS writer (for Windows), or pre and post scripts (for Linux) to achieve application consistency.

VM restore points are organized into restore point collections. A restore point collection is an Azure Resource Management resource that contains the restore points for a specific VM. If you want to utilize ARM templates for creating restore points and restore point collections, visit the public [Virtual-Machine-Restore-Points](#) repository on GitHub.

The following image illustrates the relationship between restore point collections, VM restore points, and disk restore points.



VM restore points are incremental. The first restore point stores a full copy of all disks attached to the VM. For each successive restore point for a VM, only the incremental changes to your disks are backed up. To reduce your costs, you can optionally exclude any disk when creating a restore point for your VM.

## Restore points for VMs inside Virtual Machine Scale Set and Availability Set (AvSet)

Currently, restore points can only be created in one VM at a time, that is, you cannot create a single restore point across multiple VMs. Due to this limitation, we currently support creating restore points for individual VMs with

a Virtual Machine Scale Set and Availability Set. If you want to back up your entire Virtual Machine Scale Set instance or your Availability Set instance, you must individually create restore points for all the VMs that are part of the instance.

**NOTE**

Virtual Machine Scale Set with Unified orchestration is not supported by restore points. You cannot create restore points of VMs inside a Virtual Machine Scale Set with Unified orchestration.

## Limitations

- Restore points are supported only for managed disks.
- Ultra-disks, Ephemeral OS disks, and Shared disks are not supported.
- Restore points APIs require an API of version 2021-03-01 or later.
- A maximum of 500 VM restore points can be retained at any time for a VM, irrespective of the number of restore point collections.
- Concurrent creation of restore points for a VM is not supported.
- Movement of Virtual Machines (VM) between Resource Groups (RG), or Subscriptions is not supported when the VM has restore points. Moving the VM between Resource Groups or Subscriptions will not update the source VM reference in the restore point and will cause a mismatch of ARM IDs between the actual VM and the restore points.

**NOTE**

Public preview of cross-region creation and copying of VM restore points is available, with the following limitations:

- Private links are not supported when copying restore points across regions or creating restore points in a region other than the source VM.
- Customer-managed key encrypted restore points, when copied to a target region or created directly in the target region are created as platform-managed key encrypted restore points.

## Troubleshoot VM restore points

Most common restore points failures are attributed to the communication with the VM agent and extension, and can be resolved by following the troubleshooting steps listed in the [troubleshooting](#) article.

## Next steps

- [Create a VM restore point](#).
- [Learn more](#) about Backup and restore options for virtual machines in Azure.

# Quickstart: Create VM restore points using APIs

9/21/2022 • 2 minutes to read • [Edit Online](#)

You can protect your data by taking backups at regular intervals. Azure VM restore point APIs are a lightweight option you can use to implement granular backup and retention policies. VM restore points support application consistency for VMs running Windows operating systems and support file system consistency for VMs running Linux operating system.

You can use the APIs to create restore points for your source VM in either the same region, or in other regions. You can also copy existing VM restore points between regions.

## Prerequisites

- [Learn more](#) about the requirements for a VM restore point.
- Consider the [limitations](#) before creating a restore point.

## Create VM restore points

The following sections outline the steps you need to take to create VM restore points with the Azure Compute REST APIs.

You can find more information in the [Restore Points](#), [PowerShell](#), and [Restore Point Collections API](#) documentation.

### Step 1: Create a VM restore point collection

Before you create VM restore points, you must create a restore point collection. A restore point collection holds all the restore points for a specific VM. Depending on your needs, you can create VM restore points in the same region as the VM, or in a different region. To create a restore point collection, call the restore point collection's Create or Update API.

- If you're creating restore point collection in the same region as the VM, then specify the VM's region in the location property of the request body.
- If you're creating the restore point collection in a different region than the VM, specify the target region for the collection in the location property, but also specify the source restore point collection ARM resource ID in the request body.

To create a restore point collection, call the restore point collection's [Create or Update](#) API.

### Step 2: Create a VM restore point

After you create the restore point collection, the next step is to create a VM restore point within the restore point collection. For more information about restore point creation, see the [Restore Points - Create API](#) documentation.

#### TIP

To save space and costs, you can exclude any disk from either local region or cross-region VM restore points. To exclude a disk, add its identifier to the `excludeDisks` property in the request body.

### Step 3: Track the status of the VM restore point creation

Restore point creation in your local region will be completed within a few seconds. Scenarios, which involve the

creation of cross-region restore points will take considerably longer. To track the status of the creation operation, follow the guidance in [Get restore point copy or replication status](#). This is only applicable for scenarios where the restore points are created in a different region than the source VM.

## Get restore point copy or replication status

Creation of a cross-region VM restore point is a long running operation. The VM restore point can be used to restore a VM only after the operation is completed for all disk restore points. To track the operation's status, call the [Restore Point - Get](#) API on the target VM restore point and include the `instanceView` parameter. The return will include the percentage of data that has been copied at the time of the request.

During restore point creation, the `ProvisioningState` will appear as `Creating` in the response. If creation fails, `ProvisioningState` is set to `Failed`.

## Next steps

- [Learn more](#) about managing restore points.
- Create restore points using the [Azure portal](#), [CLI](#), or [PowerShell](#).
- [Learn more](#) about Backup and restore options for virtual machines in Azure.

# Manage VM restore points

9/21/2022 • 3 minutes to read • [Edit Online](#)

This article explains how to copy and restore a VM from a VM restore point and track the progress of the copy operation. This article also explains how to create a disk from a disk restore point and to create a shared access signature for a disk.

## Copy a VM restore point between regions

The VM restore point APIs can be used to restore a VM in a different region than the source VM. Use the following steps:

### Step 1: Create a destination VM restore point collection

To copy an existing VM restore point from one region to another, your first step is to create a restore point collection in the target or destination region. To do this, reference the restore point collection from the source region as detailed in [Create a VM restore point collection](#).

### Step 2: Create the destination VM restore point

After the restore point collection is created, trigger the creation of a restore point in the target restore point collection. Ensure that you've referenced the restore point in the source region that you want to copy and specified the source restore point's identifier in the request body. The source VM's location is inferred from the target restore point collection in which the restore point is being created. See the [Restore Points - Create API](#) documentation to create a `RestorePoint`.

### Step 3: Track copy status

To track the status of the copy operation, follow the guidance in the [Get restore point copy or replication status](#) section below. This is only applicable for scenarios where the restore points are copied to a different region than the source VM.

## Get restore point copy or replication status

Creation of a cross-region VM restore point is a long running operation. The VM restore point can be used to restore a VM only after the operation is completed for all disk restore points. To track the operation's status, call the [Restore Point - Get](#) API on the target VM restore point and include the `instanceView` parameter. The return will include the percentage of data that has been copied at the time of the request.

During restore point creation, the `ProvisioningState` will appear as `Creating` in the response. If creation fails, `ProvisioningState` is set to `Failed`.

## Create a disk using disk restore points

You can use the VM restore points APIs to restore a VM disk, which can then be used to create a new VM. Use the following steps:

### Step 1: Retrieve disk restore point identifiers

Call the [Restore Point Collections - Get](#) API on the restore point collection to get access to associated restore points and their IDs. Each VM restore point will in turn contain individual disk restore point identifiers.

### Step 2: Create a disk

After you have the list of disk restore point IDs, you can use the [Disks - Create Or Update](#) API to create a disk

from the disk restore points.

## Restore a VM with a restore point

To restore a full VM from a VM restore point, you must restore individual disks from each disk restore point. This process is described in the [Create a disk](#) section. After you restore all the disks, create a new VM and attach the restored disks to the new VM. You can also use the [ARM template](#) to restore a full VM along with all the disks.

## Get a shared access signature for a disk

To create a Shared Access Signature (SAS) for a disk within a VM restore point, pass the ID of the disk restore points via the `BeginGetAccess` API. If no active SAS exists on the restore point snapshot, a new SAS is created. The new SAS URL is returned in the response. If an active SAS already exists, the SAS duration is extended, and the pre-existing SAS URL is returned in the response.

For more information about granting access to snapshots, see the [Grant Access](#) API documentation.

## Next steps

[Learn more](#) about Backup and restore options for virtual machines in Azure.

# Create virtual machine restore points using Azure CLI

9/21/2022 • 2 minutes to read • [Edit Online](#)

You can protect your data and guard against extended downtime by creating [VM restore points](#) at regular intervals. You can create VM restore points, and [exclude disks](#) while creating the restore point, using Azure CLI. Azure CLI is used to create and manage Azure resources using command line or scripts. Alternatively, you can create VM restore points using the [Azure portal](#) or using [PowerShell](#).

The [az restore-point](#) module is used to create and manage restore points from the command line or in scripts.

In this tutorial, you learn how to:

- [Create a VM restore point collection](#)
- [Create a VM restore point](#)
- [Track the progress of Copy operation](#)
- [Restore a VM](#)

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the [az login](#) command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run [az version](#) to find the version and dependent libraries that are installed. To upgrade to the latest version, run [az upgrade](#).
- Learn more about the [support requirements](#) and [limitations](#) before creating a restore point.

## Step 1: Create a VM restore point collection

Use the [az restore-point collection create](#) command to create a VM restore point collection, as shown below:

```
az restore-point collection create --location "norwayeast" --source-id "/subscriptions/{subscription-id}/resourceGroups/ExampleRg/providers/Microsoft.Compute/virtualMachines/ExampleVM" --tags myTag1="tagValue1" --resource-group "ExampleRg" --collection-name "ExampleRpc"
```

## Step 2: Create a VM restore point

Create a VM restore point with the [az restore-point create](#) command as follows:

```
az restore-point create --resource-group "ExampleRg" --collection-name "ExampleRpc" --name "ExampleRp"
```

### Exclude disks when creating a restore point

Exclude the disks that you do not want to be a part of the restore point with the `--exclude-disks` parameter, as follows:

```
az restore-point create --exclude-disks "/subscriptions/{subscription-id}/resourceGroups/ExampleRg/providers/Microsoft.Compute/disks/ExampleDisk1" --resource-group "ExampleRg" --collection-name "ExampleRpc" --name "ExampleRp"
```

## Step 3: Track the status of the VM restore point creation

Use the [az restore-point show](#) command to track the progress of the VM restore point creation.

```
az restore-point show --resource-group "ExampleRg" --collection-name "ExampleRpc" --name "ExampleRp"
```

## Restore a VM from VM restore point

To restore a VM from a VM restore point, first restore individual disks from each disk restore point. You can also use the [ARM template](#) to restore a full VM along with all the disks.

```
# Create Disks from disk restore points
$osDiskRestorePoint = az restore-point show --resource-group "ExampleRg" --collection-name "ExampleRpc" --name "ExampleRp" --query "sourceMetadata.storageProfile.dataDisks[0].diskRestorePoint.id"
$dataDisk1RestorePoint = az restore-point show --resource-group "ExampleRg" --collection-name "ExampleRpcTarget" --name "ExampleRpTarget" --query "sourceMetadata.storageProfile.dataDisks[0].diskRestorePoint.id"
$dataDisk2RestorePoint = az restore-point show --resource-group "ExampleRg" --collection-name "ExampleRpcTarget" --name "ExampleRpTarget" --query "sourceMetadata.storageProfile.dataDisks[0].diskRestorePoint.id"

az disk create --resource-group "ExampleRg" --name "ExampleOSDisk" --sku Premium_LRS --size-gb 128 --source $osDiskRestorePoint

az disk create --resource-group "ExampleRg" --name "ExampleDataDisk1" --sku Premium_LRS --size-gb 128 --source $dataDisk1RestorePoint

az disk create --resource-group "ExampleRg" --name "ExampleDataDisk1" --sku Premium_LRS --size-gb 128 --source $dataDisk2RestorePoint
```

Once you have created the disks, [create a new VM](#) and [attach these restored disks](#) to the newly created VM.

## Next steps

[Learn more](#) about Backup and restore options for virtual machines in Azure.

# Create virtual machine restore points using PowerShell

9/21/2022 • 2 minutes to read • [Edit Online](#)

## NOTE

To interact with Azure, the Azure Az PowerShell module is recommended. See [Install Azure PowerShell](#) to get started. To learn how to migrate to the Az PowerShell module, see [Migrate Azure PowerShell from AzureRM to Az](#).

You can create Virtual Machine restore points using PowerShell scripts. The [Azure PowerShell Az](#) module is used to create and manage Azure resources from the command line or in scripts.

You can protect your data and guard against extended downtime by creating [VM restore points](#) at regular intervals. This article shows you how to create VM restore points, and [exclude disks](#) from the restore point, using the [Az.Compute](#) module. Alternatively, you can create VM restore points using the [Azure CLI](#) or in the [Azure portal](#).

In this tutorial, you learn how to:

- [Create a VM restore point collection](#)
- [Create a VM restore point](#)
- [Track the progress of Copy operation](#)
- [Restore a VM](#)

## Prerequisites

- Learn more about the [support requirements](#) and [limitations](#) before creating a restore point.

## Step 1: Create a VM restore point collection

Use the [New-AzRestorePointCollection](#) cmdlet to create a VM restore point collection.

```
New-AzRestorePointCollection -ResourceGroupName ExampleRG -Name ExampleRPC -VmId  
"/subscriptions/{SubscriptionId}/resourcegroups/  
ExampleRG/providers/microsoft.compute/virtualmachines/Example-vm-1" -Location "WestEurope"
```

## Step 2: Create a VM restore point

Create a VM restore point with the [New-AzRestorePoint](#) cmdlet as shown below:

```
New-AzRestorePoint -ResourceGroupName ExampleRG -RestorePointCollectionName ExampleRPC -Name ExampleRP
```

### Exclude disks from the restore point

Exclude certain disks that you do not want to be a part of the restore point with the `-DisksToExclude` parameter, as follows:

```
New-AzRestorePoint -ResourceGroupName ExampleRG -RestorePointCollectionName ExampleRPC -Name ExampleRP -  
DisksToExclude "/subscriptions/{SubscriptionId}/resourcegroups/  
ExampleRG/providers/Microsoft.Compute/disks/example-vm-1-data_disk_1"
```

## Step 3: Track the status of the VM restore point creation

You can track the progress of the VM restore point creation using the [Get-AzRestorePoint](#) cmdlet, as follows:

```
Get-AzRestorePoint -ResourceGroupName ExampleRG -RestorePointCollectionName ExampleRPC -Name ExampleRP
```

## Restore a VM from VM restore point

To restore a VM from a VM restore point, first restore individual disks from each disk restore point. You can also use the [ARM template](#) to restore a full VM along with all the disks.

```
# Create Disks from disk restore points  
$restorePoint = Get-AzRestorePoint -ResourceGroupName ExampleRG -RestorePointCollectionName ExampleRPC -Name ExampleRP  
  
$osDiskRestorePoint = $restorePoint.SourceMetadata.StorageProfile.OsDisk.DiskRestorePoint.Id  
$dataDisk1RestorePoint = $restorePoint.sourceMetadata.storageProfile.dataDisks[0].diskRestorePoint.id  
$dataDisk2RestorePoint = $restorePoint.sourceMetadata.storageProfile.dataDisks[1].diskRestorePoint.id  
  
New-AzDisk -DiskName "ExampleOSDisk" (New-AzDiskConfig -Location eastus -CreateOption Restore -  
SourceResourceId $osDiskRestorePoint) -ResourceGroupName ExampleRg  
  
New-AzDisk -DiskName "ExampleDataDisk1" (New-AzDiskConfig -Location eastus -CreateOption Restore -  
SourceResourceId $dataDisk1RestorePoint) -ResourceGroupName ExampleRg  
  
New-AzDisk -DiskName "ExampleDataDisk2" (New-AzDiskConfig -Location eastus -CreateOption Restore -  
SourceResourceId $dataDisk2RestorePoint) -ResourceGroupName ExampleRg
```

After you create the disks, [create a new VM](#) and [attach these restored disks](#) to the newly created VM.

## Next steps

[Learn more](#) about Backup and restore options for virtual machines in Azure.

# Create virtual machine restore points using Azure portal

9/21/2022 • 2 minutes to read • [Edit Online](#)

You can create virtual machine restore points through the Azure portal. You can protect your data and guard against extended downtime by creating [VM restore points](#) at regular intervals. This article shows you how to create VM restore points using the Azure portal. Alternatively, you can create VM restore points using the [Azure CLI](#) or using [PowerShell](#).

In this tutorial, you learn how to:

- [Create a VM restore point collection](#)
- [Create a VM restore point](#)
- [Track the progress of Copy operation](#)
- [Restore a VM](#)

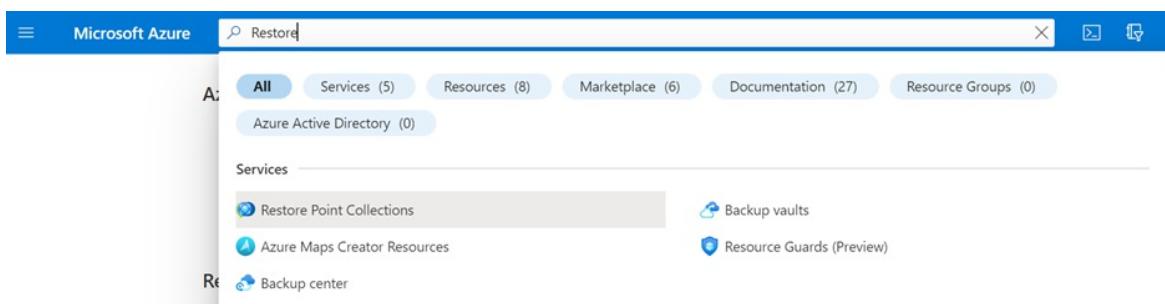
## Prerequisites

- Learn more about the [support requirements](#) and [limitations](#) before creating a restore point.

## Step 1: Create a VM restore point collection

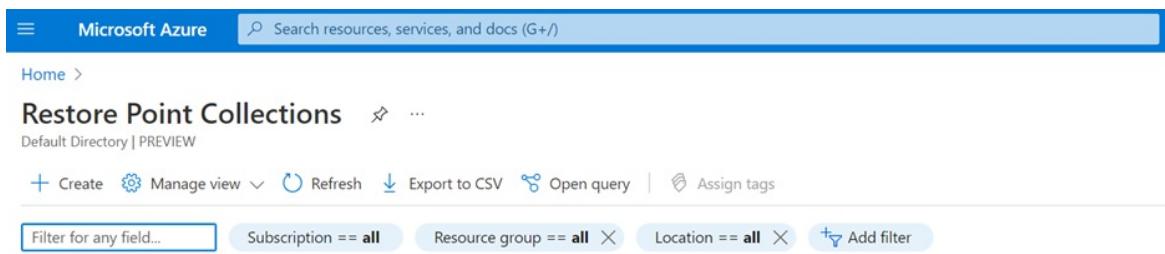
Use the following steps to create a VM restore points collection:

1. Sign in to the [Azure portal](#).
2. In the Search box, enter **Restore Point Collections**.



The screenshot shows the Microsoft Azure portal interface. At the top, there's a search bar with the word 'Restore'. Below the search bar, a navigation menu includes 'All', 'Services (5)', 'Resources (8)', 'Marketplace (6)', 'Documentation (27)', and 'Resource Groups (0)'. Under the 'Services' heading, there are several items: 'Restore Point Collections' (which is highlighted with a gray background), 'Backup vaults', 'Azure Maps Creator Resources', and 'Resource Guards (Preview)'. At the bottom of the list, there's a 'Backup center' item.

3. Select **+ Create** to create a new Restore Point Collection.



The screenshot shows the 'Restore Point Collections' blade in the Azure portal. At the top, there's a header with 'Home > Restore Point Collections' and a 'PREVIEW' note. Below the header, there are buttons for '+ Create', 'Manage view', 'Refresh', 'Export to CSV', 'Open query', and 'Assign tags'. There are also filters for 'Subscription', 'Resource group', and 'Location'. The main area displays a table of existing restore point collections, each with a name like 'Windows VM Backup' and a status like 'Success'.

4. Enter the details and select the VM for which you want to create a restore point collection.

## Create a restore point collection

Basics    Restore point    Tags    Review + create

A virtual machine restore point collection contains restore points specific to a virtual machine and each restore point contains disk restore points for each included disk. (if the VM is shutdown) or application consistent snapshot for all managed disks attached to a virtual machine.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription ⓘ \*

Visual Studio Enterprise Subscription

Resource group ⓘ \*

cloud-shell-storage-centralindia

[Create new](#)

### Instance details

Name ⓘ \*

Region ⓘ \*

(US) East US

Source virtual machine ⓘ \*

Select a virtual machine

There are no Virtual machines in the selected subscription and region.

[Review + create](#)

[< Previous](#)

[Next : Restore point >](#)

5. Select **Next: Restore Point** to create your first restore point or select **Review + Create** to create an empty restore point collection.

## Create a restore point collection

Validation passed

Basics    Restore point    Tags    [Review + create](#)

### Basics

Subscription	Visual Studio Enterprise Subscription
Resource group	RPC-Crash-Private
Restore point collection name	Test-RPC-1
Region	East US 2 EUAP
Source virtual machine	test-crash-vm-std-os
Restore point name	Test-RPC-1-rp-2022-06-22-13-30-08

### Disk

Disk count	2
Disk types	1 OS disk, 1 data disks

[Create](#)

[< Previous](#)

[Next >](#)

[Download a template for automation](#)

## Step 2: Create a VM restore point

Use the following steps to create a VM restore point:

1. Navigate to the restore point collection where you want to create restore points and select + **Create a restore point** to create new restore point for the VM.

The screenshot shows the Azure portal interface for a 'Restore Point Collection' named 'Test-RPC-1'. The left sidebar has sections for Overview, Activity log, Access control (IAM), Tags, Settings, Restore Points, Properties, Locks, Automation, Tasks (preview), Export template, Support + troubleshooting, and New Support Request. The 'Overview' tab is selected. The main content area shows 'Essentials' details: Resource group (move) : eastus2euap, Location (move) : eastus2euap, Subscription (move) : Visual Studio Enterprise Subscription, Subscription ID : (redacted), and Status : Succeeded. Below this, there's a 'Tags (edit)' section with a link to 'Click here to add tags'. A 'Get started' button is followed by a 'Restore points' button, which is underlined, indicating it's the active section. A search bar shows 'Showing 1 of 1 restore points'. Under the search bar are 'Delete' and 'Restore points' buttons. A single restore point is listed with a checkbox next to it and the name 'Test-RPC-1-rp-2022-06-22-13-30-08'.

2. Enter a name for the restore point and other required details and select **Next: Disks >**.

## Create a restore point ...

[Basics](#) [Disks](#) [Review + create](#)

Create a restore point to store virtual machine configurations and point-in-time crash (if the VM is shutdown) or application consistent snapshot for all managed disks attached to a virtual machine.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription ⓘ *	Visual Studio Enterprise Subscription
Resource group *	

### Instance details

Restore point name ⓘ *	Test-RPC-1-rp-2022-6-22-13-34-59
Restore point collection ⓘ *	Test-RPC-1
Region ⓘ *	eastus2euap
Source virtual machine ⓘ *	

[Review + create](#)

< Previous

Next : Disks >

### 3. Select the disks to be included in the restore point.

## Create a restore point ...

X

[Basics](#) [Disks](#) [Review + create](#)

Select the disks from the virtual machine that you want to include in the restore point collection. You won't be able to make changes, such as including or excluding additional disks, once the restore point is created.

#### OS disk

Disks	LUN	Storage account type	Size(GiB)
<input checked="" type="checkbox"/> Test-Crash-VM-STD-OS_OsDisk_1_2bd8252cea704e05a320570d3c1409d4		Standard_LRS	30

#### Data disk

Disks	LUN	Storage account type	Size(GiB)
<input type="checkbox"/> Test-Crash-VM-STD-Data_Disk_1	0	StandardSSD_LRS	4

[Review + create](#)

< Previous

Next : Review + create >

### 4. Select **Review + create** to validate the settings. Once validation is completed, select **Create** to create the restore point.

## Create a restore point

✓ Validation passed

Basics Disks **Review + create**

### Basics

Subscription	Visual Studio Enterprise Subscription
Resource group	
Restore point collection name	Test-RPC-1
Region	eastus2euap
Source virtual machine	
Restore point name	Test-RPC-1-rp-2022-6-22-13-34-59

### Disks

Disk count	1
Disk types	1 OS disk, 0 data disks

**Create** < Previous Next > Download a template for automation

## Step 3: Track the status of the VM restore point creation

1. Select the notification to track the progress of the restore point creation.

Home >  
RestorePoint\_Test-RPC-1-rp-2022-6-22-13-34-59\_1655885363222 | Overview ...

Deployment

Search (Ctrl + /) Delete Cancel Redeploy Refresh

**Deployment is in progress**

Deployment name: RestorePoint\_Test-RPC-1-rp-2022-6-22-13-34-59\_1655885363222  
Start time: 6/22/2022, 1:41:32 PM  
Subscription: Visual Studio Enterprise Subscription  
Resource group: eastus2euap

Deployment details (Download)

Resource	Type	Status	Operation details
No results.			

## Restore a VM from a restore point

To restore a VM from a VM restore point, first restore individual disks from each disk restore point. You can also use the [ARM template](#) to restore a VM along with all the disks.

1. Select **Create a disk from a restore point** to restore a disk from a disk restore point. Do this for all the disks that you want to restore.

**Test-RPC-1-rp-2022-6-22-13-34-59 (Test-RPC-1/Test-RPC-1-rp-2022-6-22-13-34-59)** ⋮

Restore Point

Search (Ctrl+ /) <> Delete Refresh

Overview

Essentials

Resource group : Test-RPC-1  
Subscription : Visual Studio Enterprise Subscription  
Subscription ID :  
Status : Succeeded  
Create time : 6/22/2022, 1:41:37 PM

Restore point collection : Test-RPC-1  
Included disks : 1  
Consistency mode : CrashConsistent

JSON View

Virtual machine metadata

Properties

Locks

Automation

Tasks (preview)

Export template

Support + troubleshooting

New Support Request

Disks

Show 1 of 1 disks

	Size(GiB)	Storage account type	Restore disk
<input type="checkbox"/> Test-Crash-VM-STD-OS_OsDisk_1_2bd8252cea704e05a320570d3c1409d4			<a href="#">Create disk from a restore point</a>

2. Enter the details in the **Create a managed disk** dialog to create disks from the restore points. Once the disks are created, [create a new VM](#) and [attach these restored disks](#) to the newly created VM.

## Create a managed disk ⋮

Basics Encryption Networking Advanced Tags Review + create

Select the disk type and size needed for your workload. Azure disks are designed for 99.999% availability. Azure managed disks encrypt your data at rest, by default, using Storage Service Encryption. [Learn more about disks](#).

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription ⓘ	Visual Studio Enterprise Subscription
Resource group * ⓘ	<input type="text"/>
	<a href="#">Create new</a>

### Disk details

Disk name * ⓘ	<input type="text"/>
Region ⓘ	(US) East US
Availability zone	None
Source type ⓘ	Disk restore point
Disk restore point ⓘ	<p>Restore point collection: Test-RPC-1            Restore point: Test-RPC-1-rp-2022-6-22-13-34-59            Disk: Test-Crash-VM-STD-OS_OsDisk_1_2bd8252cea704e05a320570d3c1409d4  <a href="#">Select a disk restore point</a></p>
Size * ⓘ	<p>1024 GiB            Premium SSD LRS  <a href="#">Change size</a></p>

[Review + create](#)

< Previous

Next : Encryption >

## Next steps

[Learn more](#) about Backup and restore options for virtual machines in Azure.

# Support matrix for VM restore points

9/21/2022 • 3 minutes to read • [Edit Online](#)

This article summarizes the support matrix and limitations of using [VM restore points](#).

## VM restore points support matrix

The following table summarizes the support matrix for VM restore points.

SCENARIOS	SUPPORTED BY VM RESTORE POINTS
VMs using Managed disks	Yes
VMs using unmanaged disks	No
VMs using Ultra Disks	No. Exclude these disks and create a VM restore point.
VMs using Ephemeral OS Disks	No. Exclude these disks and create a VM restore point.
VMs using shared disks	No. Exclude these disks and create a VM restore point.
VMs with extensions	Yes
VMs with trusted launch	Yes
Confidential VMs	Yes
Generation 2 VMs (UEFI boot)	Yes
VMs with NVMe disks (Storage optimized - Lsv2-series)	Yes
VMs in Proximity placement groups	Yes
VMs in an availability set	Yes. You can create VM restore points for individual VMs within an availability set. You need to create restore points for all the VMs within an availability set to protect an entire availability set instance.
VMs inside VMSS with uniform orchestration	No
VMs inside VMSS with flexible orchestration	Yes. You can create VM restore points for individual VMs within the virtual machine scale set flex. However, you need to create restore points for all the VMs within the virtual machine scale set flex to protect an entire virtual machine scale set flex instance.
Spot VMs (Low priority VMs)	Yes
VMs with dedicated hosts	Yes

SCENARIOS	SUPPORTED BY VM RESTORE POINTS
VMs with Host caching enabled	Yes
VMs created from marketplace images	Yes
VMs created from custom images	Yes
VM with HUB (Hybrid Use Benefit) license	Yes
VMs migrated from on-prem using Azure Migrate	Yes
VMs with RBAC policies	Yes
Temporary disk in VMs	Yes. You can create VM restore point for VMs with temporary disks. However, the restore points created don't contain the data from the temporary disks.
VMs with standard HDDs	Yes
VMs with standard SSDs	Yes
VMs with premium SSDs	Yes
VMs with ZRS disks	Yes
VMs with server-side encryption using service-managed keys	Yes
VMs with server-side encryption using customer-managed keys	Yes
VMs with double encryption at rest	Yes
VMs with Host based encryption enabled with PMK/CMK/Double encryption	Yes
VMs with ADE (Azure Disk Encryption)	Yes
VMs using Accelerated Networking	Yes
Frequency supported	Three hours for app consistent restore points. One hour for <a href="#">crash consistent restore points (preview)</a>

## Operating system support

### Windows

The following Windows operating systems are supported when creating restore points for Azure VMs running on Windows.

- Windows 10 Client (64 bit only)
- Windows Server 2022 (Datacenter/Datacenter Core/Standard)
- Windows Server 2019 (Datacenter/Datacenter Core/Standard)

- Windows Server 2016 (Datacenter/Datacenter Core/Standard)
- Windows Server 2012 R2 (Datacenter/Standard)
- Windows Server 2012 (Datacenter/Standard)
- Windows Server 2008 R2 (RTM and SP1 Standard)
- Windows Server 2008 (64 bit only)

Restore points don't support 32-bit operating systems.

## Linux

For Azure VM Linux VMs, restore points support the list of Linux [distributions endorsed by Azure](#). Note the following:

- Restore points don't support Core OS Linux.
- Restore points don't support 32-bit operating systems.
- Other bring-your-own Linux distributions might work as long as the [Azure VM agent for Linux](#) is available on the VM, and as long as Python is supported.
- Restore points don't support a proxy-configured Linux VM if it doesn't have Python version 2.7 or higher installed.
- Restore points don't back up NFS files that are mounted from storage, or from any other NFS server, to Linux or Windows machines. It only backs up disks that are locally attached to the VM.

## Other limitations

- Restore points are supported only for managed disks.
- Ultra-disks, Ephemeral OS disks, and Shared disks aren't supported.
- Restore points APIs require an API of version 2021-03-01 or later.
- A maximum of 500 VM restore points can be retained at any time for a VM, irrespective of the number of restore point collections.
- Concurrent creation of restore points for a VM isn't supported.
- Movement of Virtual Machines (VM) between Resource Groups (RG), or Subscriptions isn't supported when the VM has restore points. Moving the VM between Resource Groups or Subscriptions won't update the source VM reference in the restore point and will cause a mismatch of ARM processor IDs between the actual VM and the restore points.

### NOTE

Public preview of cross-region creation and copying of VM restore points is available, with the following limitations:

- Private links aren't supported when copying restore points across regions or creating restore points in a region other than the source VM.
- Customer-managed key encrypted restore points, when copied to a target region or created directly in the target region are created as platform-managed key encrypted restore points.
- No portal support for cross region copy and cross region creation of restore points

## Next steps

- Learn how to create VM restore points using [CLI](#), [Azure portal](#), and [PowerShell](#).

# Troubleshoot restore point failures: Issues with the agent or extension

9/21/2022 • 19 minutes to read • [Edit Online](#)

This article provides troubleshooting steps that can help you resolve restore point errors related to communication with the VM agent and extension.

If your Azure issue is not addressed in this article, visit the Azure forums on [Microsoft Q & A and Stack Overflow](#). You can post your issue in these forums, or post to [@AzureSupport on Twitter](#). You also can submit an Azure support request. To submit a support request, on the [Azure support](#) page, select **Get support**.

## Step-by-step guide to troubleshoot restore point failures

Most common restore point failures can be resolved by following the troubleshooting steps listed below:

### Step 1: Check the health of Azure VM

- Ensure Azure VM provisioning state is 'Running': If the [VM provisioning state](#) is in the **Stopped/Deallocated/Updating** state, it interferes with the restore point operation. In the Azure portal, go to **Virtual Machines > Overview** and ensure the VM status is **Running** and retry the restore point operation.
- Review pending OS updates or reboots: Ensure there are no pending OS updates or pending reboots on the VM.

### Step 2: Check the health of Azure VM Guest Agent service

Ensure Azure VM Guest Agent service is started and up-to-date:

- On a Windows VM:
  - Navigate to **services.msc** and ensure **Windows Azure VM Guest Agent service** is up and running. Also, ensure the [latest version](#) is installed. [Learn more](#).
  - The Azure VM Agent is installed by default on any Windows VM deployed from an Azure Marketplace image from the portal, PowerShell, Command Line Interface, or an Azure Resource Manager template. A [manual installation of the Agent](#) may be necessary when you create a custom VM image that's deployed to Azure.
  - Review the support matrix to check if VM runs on the [supported Windows operating system](#).
- On Linux VM,
  - Ensure the Azure VM Guest Agent service is running by executing the command `ps -e`. Also, ensure the [latest version](#) is installed. [Learn more](#).
  - Ensure the [Linux VM agent dependencies on system packages](#) have the supported configuration. For example: Supported Python version is 2.6 and above.
  - Review the support matrix to check if VM runs on the [supported Linux operating system](#).

### Step 3: Check the health of Azure VM Extension

- Ensure all Azure VM Extensions are in 'provisioning succeeded' state: If any extension is in a failed state, then it can interfere with the restore point operation.
  - In the Azure portal, go to **Virtual machines > Settings > Extensions > Extensions status** and check if all the extensions are in **provisioning succeeded** state.
  - Ensure all [extension issues](#) are resolved and retry the restore point operation.
- Ensure COM+ System Application is up and running. Also, the **Distributed Transaction Coordinator**

service should be running as **Network Service account**.

Follow the troubleshooting steps in [troubleshoot COM+ and MSDTC issues](#) in case of issues.

#### Step 4: Check the health of Azure VM Snapshot Extension

Restore points use the VM Snapshot Extension to take an application consistent snapshot of the Azure virtual machine. Restore points install the extension as part of the first restore point creation operation.

- **Ensure VMSnapshot extension isn't in a failed state:** Follow the steps in [Troubleshooting](#) to verify and ensure the Azure VM snapshot extension is healthy.
- **Check if antivirus is blocking the extension:** Certain antivirus software can prevent extensions from executing.

At the time of the restore point failure, verify if there are log entries in **Event Viewer Application logs** with *faulting application name: IaaSBcdrExtension.exe*. If you see entries, the antivirus configured in the VM could be restricting the execution of the VMSnapshot extension. Test by excluding the following directories in the antivirus configuration and retry the restore point operation.

- C:\Packages\Plugins\Microsoft.Azure.RecoveryServices.VMSnapshot
- C:\WindowsAzure\Logs\Plugins\Microsoft.Azure.RecoveryServices.VMSnapshot

- **Check if network access is required:** Extension packages are downloaded from the Azure Storage extension repository and extension status uploads are posted to Azure Storage. [Learn more](#).
  - If you're on a non-supported version of the agent, you need to allow outbound access to Azure storage in that region from the VM.
  - If you've blocked access to 168.63.129.16 using the guest firewall or with a proxy, extensions will fail regardless of the above. Ports 80, 443, and 32526 are required. [Learn more](#).
- **Ensure DHCP is enabled inside the guest VM:** This is required to get the host or fabric address from DHCP for the restore point to work. If you need a static private IP, you should configure it through the [Azure portal](#), or [PowerShell](#) and make sure the DHCP option inside the VM is enabled. [Learn more](#).
- **Ensure the VSS writer service is up and running:** Follow these steps to [troubleshoot VSS writer issues](#).

## Common issues

### DiskRestorePointUsedByCustomer - There is an active shared access signature outstanding for disk restore point

**Error code:** DiskRestorePointUsedByCustomer

**Error message:** There is an active shared access signature outstanding for disk restore point. Call EndGetAccess before deleting the restore point.

You can't delete a restore point if there are active Shared Access Signatures (SAS) on any of the underlying disk restore points. End the shared access on the disk restore points and retry the operation.

### OperationNotAllowed - Changes were made to the Virtual Machine while the operation 'Create Restore Point' was in progress.

**Error code:** OperationNotAllowed

**Error message:** Changes were made to the Virtual Machine while the operation 'Create Restore Point' was in progress. Operation Create Restore Point cannot be completed at this time. Please try again later.

Restore point creation fails if there are changes being made in parallel to the VM model, for example, a new disk being attached or an existing disk being detached. This is to ensure data integrity of the restore point that is created. Retry creating the restore point once the VM model has been updated.

## **OperationNotAllowed - Operation 'Create Restore Point' is not allowed as disk(s) have not been allocated successfully.**

**Error code:** OperationNotAllowed

**Error message:** Operation 'Create Restore Point' is not allowed as disk(s) have not been allocated successfully. Please exclude these disk(s) using excludeDisks property and retry.

If any one of the disks attached to the VM isn't allocated properly, the restore point fails. You must exclude these disks before triggering creation of restore points for the VM. If you're using ARM processor API to create a restore point, to exclude a disk, add its identifier to the excludeDisks property in the request body. If you're using [CLI](#), [PowerShell](#), or [Portal](#), set the respective parameters.

## **OperationNotAllowed - Creation of Restore Point of a Virtual Machine with Shared disks is not supported.**

**Error code:** VMRestorePointClientError

**Error message:** Creation of Restore Point of a Virtual Machine with Shared disks is not supported. You may exclude this disk from the restore point via excludeDisks property.

Restore points are currently not supported for shared disks. You need to exclude these disks before triggering creation of restore point for the VM. If you are using ARM processor API to create restore point, to exclude a disk, add its identifier to the excludeDisks property in the request body. If you are using [CLI](#), [PowerShell](#), or [Portal](#), follow the respective steps.

## **VMAgentStatusCommunicationError - VM agent unable to communicate with compute service**

**Error code:** VMAgentStatusCommunicationError

**Error message:** VM has not reported status for VM agent or extensions.

The Azure VM agent might be stopped, outdated, in an inconsistent state, or not installed. These states prevent the creation of restore points.

- In the Azure portal, go to **Virtual Machines > Settings > Properties** and ensure that the **VM Status** is **Running** and **Agent status** is **Ready**. If the VM agent is stopped or is in an inconsistent state, restart the agent.
  - [Restart](#) the Guest Agent for Windows VMs.
  - [Restart](#) the Guest Agent for Linux VMs.
- In the Azure portal, go to **Virtual Machines > Settings > Extensions** and ensure all extensions are in **provisioning succeeded** state. If not, follow these [steps](#) to resolve the issue.

## **VMRestorePointInternalError - Restore Point creation failed due to an internal execution error while creating VM snapshot. Please retry the operation after some time.Internal**

**Error code:** VMRestorePointInternalError

**Error message:** Restore Point creation failed due to an internal execution error while creating VM snapshot. Please retry the operation after some time.

After you trigger a restore point operation, the compute service starts the job by communicating with the VM backup extension to take a point-in-time snapshot. Any of the following conditions might prevent the snapshot from being triggered. If the snapshot isn't triggered, restore point creation will fail. Complete the following troubleshooting steps in the order listed, and then retry your operation:

**Cause 1:** [The agent is installed in the VM, but it's unresponsive \(for Windows VMs\)](#)

**Cause 2:** [The agent installed in the VM is out of date \(for Linux VMs\)](#)

**Cause 3:** [The snapshot status can't be retrieved, or a snapshot can't be taken](#)

**Cause 4:** [VM-Agent configuration options aren't set \(for Linux VMs\)](#)

## Cause 5: Application control solution is blocking IaaSBcdrExtension.exe

This error could also occur when one of the extension failures puts the VM into provisioning failed state. If the above steps didn't resolve your issue, then do the following:

In the Azure portal, go to **Virtual Machines > Settings > Extensions** and ensure all extensions are in **provisioning succeeded** state. [Learn more](#) about Provisioning states.

- If any extension is in a failed state, it can interfere with the restore point operation. Ensure the extension issues are resolved and retry the restore point operation.
- If the VM provisioning state is in an updating state, it can interfere with the restore point operation. Ensure that it's healthy and retry the restore point operation.

## VMRestorePointClientError - Restore Point creation failed due to COM+ error.

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to COM+ error. Please restart windows service "COM+ System Application" (COMSysApp). If the issue persists, restart the VM.

Restore point operations fail if the COM+ service is not running or if there are any errors with this service. Restart the COM+ System Application, and restart the VM and retry the restore point operation.

## VMRestorePointClientError - Restore Point creation failed due to insufficient memory available in COM+ memory quota.

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to insufficient memory available in COM+ memory quota. Please restart windows service "COM+ System Application" (COMSysApp). If the issue persists, restart the VM.

Restore point operations fail if there's insufficient memory in the COM+ service. Restarting the COM+ System Application service and the VM usually frees up the memory. Once restarted, retry the restore point operation.

## VMRestorePointClientError - Restore Point creation failed due to VSS Writers in bad state.

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to VSS Writers in bad state. Restart VSS Writer services and reboot VM.

Restore point creation invokes VSS writers to flush in-memory IOs to the disk before taking snapshots to achieve application consistency. If the VSS writers are in bad state, it affects the restore point creation operation. Restart the VSS writer service and restart the VM before retrying the operation.

## VMRestorePointClientError - Restore Point creation failed due to failure in installation of Visual C++ Redistributable for Visual Studio 2012.

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to failure in installation of Visual C++ Redistributable for Visual Studio 2012. Please install Visual C++ Redistributable for Visual Studio 2012. If you are observing issues with installation or if it is already installed and you are observing this error, please restart the VM to clean installation issues.

Restore point operations require Visual C++ Redistributable for Visual Studio 2021. Download Visual C++ Redistributable for Visual Studio 2012 and restart the VM before retrying the restore point operation.

## VMRestorePointClientError - Restore Point creation failed as the maximum allowed snapshot limit of one or more disk blobs has been reached. Please delete some existing restore points of this VM and then retry.

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed as the maximum allowed snapshot limit of one or more disk blobs

has been reached. Please delete some existing restore points of this VM and then retry.

The number of restore points across the restore point collections and resource groups for a VM can't exceed 500. To create a new restore point, delete the existing restore points.

### **VMRestorePointClientError - Restore Point creation failed with the error "COM+ was unable to talk to the Microsoft Distributed Transaction Coordinator".**

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed with the error "COM+ was unable to talk to the Microsoft Distributed Transaction Coordinator".

Follow these steps to resolve this error:

- Open services.msc from an elevated command prompt
- Make sure that **Log On As** value for **Distributed Transaction Coordinator** service is set to **Network Service** and the service is running.
- If this service fails to start, reinstall this service.

### **VMRestorePointClientError - Restore Point creation failed due to inadequate VM resources.**

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to inadequate VM resources. Increase VM resources by changing the VM size and retry the operation. To resize the virtual machine, refer <https://azure.microsoft.com/blog/resize-virtual-machines/>.

Creating a restore point requires enough compute resource to be available. If you get the above error when creating a restore point, you need resize the VM and choose a higher VM size. Follow the steps in [how to resize your VM](#). Once the VM is resized, retry the restore point operation.

### **VMRestorePointClientError - Restore point creation failed due to no network connectivity on the virtual machine.**

**Error code:** VMRestorePointClientError

**Error message:** Restore Point creation failed due to no network connectivity on the virtual machine. Ensure that VM has network access. Either allowlist the Azure datacenter IP ranges or set up a proxy server for network access. For more information, see <https://go.microsoft.com/fwlink/?LinkId=800034>. If you are already using proxy server, make sure that proxy server settings are configured correctly.

After you trigger creation of restore point, the compute service starts communicating with the VM snapshot extension to take a point-in-time snapshot. Any of the following conditions might prevent the snapshot from being triggered. If the snapshot isn't triggered, a restore point failure might occur. Complete the following troubleshooting step, and then retry your operation:

[The snapshot status can't be retrieved, or a snapshot can't be taken].(#the-snapshot-status-can't-be-retrieved-or-a-snapshot-can't-be-taken)

### **VMRestorePointClientError - RestorePoint creation failed since a concurrent 'Create RestorePoint' operation was triggered on the VM.**

**Error code:** VMRestorePointClientError

**Error message:** RestorePoint creation failed since a concurrent 'Create RestorePoint' operation was triggered on the VM.

Your recent restore point creation failed because there's already an existing restore point being created. You can't create a new restore point until the current restore point is fully created. Ensure the restore point creation operation currently in progress is completed before triggering another restore point creation operation.

To check the restore points in progress, do the following steps:

1. Sign in to the Azure portal, select **All services**. Enter **Recovery Services** and select **Restore point collection**. The list of Restore point collections appears.
2. From the list of Restore point collections, select a Restore point collection in which the restore point is being created.
3. Select **Settings > Restore points** to view all the restore points. If a restore point is in progress, wait for it to complete.
4. Retry creating a new restore point.

**DiskRestorePointClientError - Keyvault associated with DiskEncryptionSet is not found.**

**Error code:** DiskRestorePointClientError

**Error message:** Keyvault associated with DiskEncryptionSet not found. The resource may have been deleted due to which Restore Point creation failed. Please retry the operation after re-creating the missing resource with the same name.

If you are creating restore points for a VM that has encrypted disks, you must ensure the keyvault where the keys are stored, is available. We use the same keys to create encrypted restore points.

**BadRequest - This request can be made with api-version '2021-03-01' or newer**

**Error code:** BadRequest

**Error message:** This request can be made with api-version '2022-03-01' or newer.

Restore points are supported only with API version 2022-03-01 or later. If you are using REST APIs to create and manage restore points, use the specified API version when calling the restore point API.

**InternalError / InternalExecutionError / InternalOperationError - An internal execution error occurred. Please retry later.**

**Error code:** InternalError / InternalExecutionError / InternalOperationError

**Error message:** An internal execution error occurred. Please retry later.

After you trigger creation of restore point, the compute service starts communicating with the VM snapshot extension to take a point-in-time snapshot. Any of the following conditions might prevent the snapshot from being triggered. If the snapshot isn't triggered, a restore point failure might occur. Complete the following troubleshooting steps in the order listed, and then retry your operation:

- Cause 1: [The agent is installed in the VM, but it's unresponsive \(for Windows VMs\)](#).
- Cause 2: [The agent installed in the VM is out of date \(for Linux VMs\)](#).
- Cause 3: [The snapshot status can't be retrieved, or a snapshot can't be taken](#).
- Cause 4: [Compute service does not have permission to delete the old restore points because of a resource group lock](#).
- Cause 5: There's an extension version/bits mismatch with the Windows version you're running, or the following module is corrupt:

C:\Packages\Plugins\Microsoft.Azure.RecoveryServices.VMSnapshot\<extension version>\iaasvmprovider.dll

To resolve this issue, check if the module is compatible with x86 (32-bit)/x64 (64-bit) version of *regsvr32.exe*, and then follow these steps:

1. In the affected VM, go to Control panel > Program and features.
2. Uninstall Visual C++ Redistributable x64 for Visual Studio 2013.
3. Reinstall Visual C++ Redistributable for Visual Studio 2013 in the VM. To install, follow these

steps:

- a. Go to the folder: C:\Packages\Plugins\Microsoft.Azure.RecoveryServices.VMSnapshot\<LatestVersion>.
- b. Search and run the vcredist2013\_x64 file to install.
4. Retry the restore point operation.

#### **OSProvisioningClientError - Restore points operation failed due to an error. For details, see restore point provisioning error Message details**

**Error code:** OSProvisioningClientError

**Error message:** OS Provisioning did not finish in the allotted time. This error occurred too many times consecutively from image. Make sure the image has been properly prepared (generalized).

This error is reported from the IaaS VM. Take necessary actions as described in the error message and retry the operation.

#### **AllocationFailed - Restore points operation failed due to an error. For details, see restore point provisioning error Message details**

**Error code:** AllocationFailed

**Error message:** Allocation failed. If you are trying to add a new VM to an Availability Set or update/resize an existing VM in an Availability Set, please note that such Availability Set allocation is scoped to a single cluster, and it is possible that the cluster is out of capacity. [Learn more](#) about improving likelihood of allocation success.

This error is reported from the IaaS VM. Take necessary actions as described in the error message and retry the operation.

## Causes and solutions

### **The agent is installed in the VM, but it's unresponsive (for Windows VMs)**

#### **Solution**

The VM agent might have been corrupted, or the service might have been stopped. Reinstalling the VM agent helps get the latest version. It also helps restart communication with the service.

1. Determine whether the Microsoft Azure Guest Agent service is running in the VM services (services.msc). Try to restart the Microsoft Azure Guest Agent service and initiate the restore point operation.
2. If the Microsoft Azure Guest Agent service isn't visible in services, in Control Panel, go to **Programs and Features** to determine whether the Microsoft Azure Guest Agent service is installed.
3. If the Microsoft Azure Guest Agent appears in **Programs and Features**, uninstall the Microsoft Azure Guest Agent.
4. Download and install the [latest version of the agent MSI](#). You must have Administrator rights to complete the installation.
5. Verify that the Microsoft Azure Guest Agent services appear in services.
6. Retry the restore point operation.

Also, verify that [Microsoft .NET 4.5 is installed](#) in the VM. .NET 4.5 is required for the VM agent to communicate with the service.

### **The agent installed in the VM is out of date (for Linux VMs)**

#### **Solution**

Most agent-related or extension-related failures for Linux VMs are caused by issues that affect an outdated VM agent. To troubleshoot this issue, follow these general guidelines:

1. Follow the instructions for [updating the Linux VM agent](#).

#### NOTE

We *strongly recommend* that you update the agent only through a distribution repository. We don't recommend downloading the agent code directly from GitHub and updating it. If the latest agent for your distribution is not available, contact distribution support for instructions on how to install it. To check for the most recent agent, go to the [Windows Azure Linux agent](#) page in the GitHub repository.

2. Ensure that the Azure agent is running on the VM by running the following command: `ps -e`

If the process isn't running, restart it by using the following commands:

- For Ubuntu: `service walinuxagent start`
- For other distributions: `service waagent start`

3. [Configure the auto restart agent](#).

4. Retry the restore point operation. If the failure persists, collect the following logs from the VM:

- `/var/lib/waagent/*.xml`
- `/var/log/waagent.log`
- `/var/log/azure/*`

If you require verbose logging for waagent, follow these steps:

1. In the `/etc/waagent.conf` file, locate the following line: **Enable verbose logging (y|n)**.
2. Change the **Logs.Verbose** value from *n* to *y*.
3. Save the change, and then restart waagent by completing the steps described earlier in this section.

#### VM-Agent configuration options are not set (for Linux VMs)

A configuration file (`/etc/waagent.conf`) controls the actions of waagent. Configuration File Options **Extensions.Enable** should be set to *y* and **Provisioning.Agent** should be set to **auto** for restore points to work. For the full list of VM-Agent Configuration File Options, see <https://github.com/Azure/WALinuxAgent#configuration-file-options>.

#### Application control solution is blocking IaaSBcdrExtension.exe

If you're running [AppLocker](#) (or another application control solution), and the rules are publisher or path based, they may block the **IaaSBcdrExtension.exe** executable from running.

#### Solution

Exclude the `/var/lib` path or the **IaaSBcdrExtension.exe** executable from AppLocker (or other application control software.)

#### The snapshot status can't be retrieved, or a snapshot can't be taken

Restore points rely on issuing a snapshot command to the underlying storage account. Restore point can fail either because it has no access to the storage account, or because the execution of the snapshot task is delayed.

#### Solution

The following conditions might cause the snapshot task to fail:

CAUSE	SOLUTION
The VM status is reported incorrectly because the VM is shut down in Remote Desktop Protocol (RDP).	If you shut down the VM in RDP, check the portal to determine whether the VM status is correct. If it's not correct, shut down the VM in the portal by using the <b>Shutdown</b> option on the VM dashboard.

CAUSE	SOLUTION
The VM can't get the host or fabric address from DHCP.	DHCP must be enabled inside the guest for restore point to work. If the VM can't get the host or fabric address from DHCP response 245, it can't download or run any extensions. If you need a static private IP, you should configure it through the <a href="#">Azure portal</a> , or <a href="#">PowerShell</a> and make sure the DHCP option inside the VM is enabled. <a href="#">Learn more</a> about setting up a static IP address with PowerShell.

## Remove lock from the recovery point resource group

1. Sign in to the [Azure portal](#).
2. Go to **All Resources**, select the restore point collection resource group.
3. In the **Settings** section, select **Locks** to display the locks.
4. To remove the lock, select **Delete**.

Lock name	Lock type	Scope	Notes
TestLock	Read-only	[Scope Icon]	

[Edit](#)  [Delete](#)

# Move a VM to another subscription or resource group

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Flexible scale sets

This article walks you through how to move a virtual machine (VM) between resource groups or subscriptions. Moving a VM between subscriptions can be handy if you created a VM in a personal subscription and now want to move it to your company's subscription.

## IMPORTANT

New resource IDs are created as part of the move. After the VM has been moved, you will need to update your tools and scripts to use the new resource IDs.

## Use the Azure CLI to move a VM

Before you can move your VM by using the Azure CLI, you need to make sure the source and destination subscriptions exist within the same tenant. To check that both subscriptions have the same tenant ID, use [az account show](#).

```
az account show --subscription mySourceSubscription --query tenantId
az account show --subscription myDestinationSubscription --query tenantId
```

If the tenant IDs for the source and destination subscriptions are not the same, you must contact [support](#) to move the resources to a new tenant.

To successfully move a VM, you need to move the VM and all its supporting resources. Use the [az resource list](#) command to list all the resources in a resource group and their IDs. It helps to pipe the output of this command to a file so you can copy and paste the IDs into later commands.

```
az resource list --resource-group "mySourceResourceGroup" --query "[].{Id:id}" --output table
```

The `table` output isn't available if you use `--interactive`. Change the output to another option like `json`.

To move a VM and its resources to another resource group, use [az resource move](#). The following example shows how to move a VM and the most common resources it requires. Use the `-ids` parameter and pass in a comma-separated list (without spaces) of IDs for the resources to move.

```

vm=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Compute/virtualMachines/myVM
nic=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Network/networkInterfaces/myNIC
nsg=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Network/networkSecurityGroups/myNSG
pip=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Network/publicIPAddresses/myPublicIPAddress
vnet=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Network/virtualNetworks/myVNet
diag=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Storage/storageAccounts/mydiagnosticstorageaccount
storage=/subscriptions/mySourceSubscriptionID/resourceGroups/mySourceResourceGroup/providers/Microsoft.Storage/storageAccounts/mystorageaccountname

az resource move \
    --ids $vm $nic $nsg $pip $vnet $storage $diag \
    --destination-group "myDestinationResourceGroup"

```

If you want to move the VM and its resources to a different subscription, add the **--destination-subscriptionId** parameter to specify the destination subscription.

When you are asked to confirm that you want to move the specified resources, enter Y to confirm.

## Use the Azure portal to move a VM to a different subscription

You can move a VM and its associated resources to a different subscription by using the Azure portal.

1. Go to the [Azure portal](#) to manage the resource group containing the VM to move. Search for and select **Resource groups**.
2. Choose the resource group containing the VM that you would like to move.
3. At the top of the page for the resource group, select **Move** and then select **Move to another subscription**. The **Move resources** page opens.
4. Select each of the resources to move. In most cases, you should move all of the related resources that are listed.
5. Select the **Subscription** where you want the VM to be moved.
6. Select an existing **Resource group**, or enter a name to have a new resource group created.
7. When you are done, select that you understand that new resource IDs will be created and that the new IDs will need to be used with the VM after it is moved, and then select **OK**.

## Use the Azure portal to move a VM to another resource group

You can move a VM and its associated resources to another resource group by using the Azure portal.

1. Go to the [Azure portal](#) to manage the resource group containing the VM to move. Search for and select **Resource groups**.
2. Choose the resource group containing the VM that you would like to move.
3. At the top of the page for the resource group, select **Move** and then select **Move to another resource group**. The **Move resources** page opens.
4. Select each of the resources to move. In most cases, you should move all of the related resources that are listed.
5. Select an existing **Resource group**, or enter a name to have a new resource group created.
6. When you are done, select that you understand that new resource IDs will be created and that the new IDs will need to be used with the VM after it is moved, and then select **OK**.

## Next steps

You can move many different types of resources between resource groups and subscriptions. For more information, see [Move resources to a new resource group or subscription](#).

# Move a Windows VM to another Azure subscription or resource group

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article walks you through how to move a Windows virtual machine (VM) between resource groups or subscriptions. Moving between subscriptions can be handy if you originally created a VM in a personal subscription and now want to move it to your company's subscription to continue your work. You do not need to stop the VM in order to move it and it should continue to run during the move.

## IMPORTANT

New resource IDs are created as part of the move. After the VM has been moved, you will need to update your tools and scripts to use the new resource IDs.

## Use the Azure portal to move a VM to a different subscription

You can move a VM and its associated resources to a different subscription by using the Azure portal.

1. Go to the [Azure portal](#) to manage the resource group containing the VM to move. Search for and select **Resource groups**.
2. Choose the resource group containing the VM that you would like to move.
3. At the top of the page for the resource group, select **Move** and then select **Move to another subscription**. The **Move resources** page opens.
4. Select each of the resources to move. In most cases, you should move all of the related resources that are listed.
5. Select the **Subscription** where you want the VM to be moved.
6. Select an existing **Resource group**, or enter a name to have a new resource group created.
7. When you are done, select that you understand that new resource IDs will be created and that the new IDs will need to be used with the VM after it is moved, and then select **OK**.

## Use the Azure portal to move a VM to another resource group

You can move a VM and its associated resources to another resource group by using the Azure portal.

1. Go to the [Azure portal](#) to manage the resource group containing the VM to move. Search for and select **Resource groups**.
2. Choose the resource group containing the VM that you would like to move.
3. At the top of the page for the resource group, select **Move** and then select **Move to another resource group**. The **Move resources** page opens.
4. Select each of the resources to move. In most cases, you should move all of the related resources that are listed.
5. Select an existing **Resource group**, or enter a name to have a new resource group created.
6. When you are done, select that you understand that new resource IDs will be created and that the new IDs will need to be used with the VM after it is moved, and then select **OK**.

## Use PowerShell to move a VM

To move a virtual machine to another resource group, you need to make sure that you also move all of the dependent resources. To get a list with the resource ID of each of these resources, use the [Get-AzResource](#) cmdlet.

```
Get-AzResource -ResourceGroupName myResourceGroup | Format-table -wrap -Property ResourceId
```

You can use the output of the previous command to create a comma-separated list of resource IDs to [Move-AzResource](#) to move each resource to the destination.

```
Move-AzResource -DestinationResourceGroupName "myDestinationResourceGroup" `  
-ResourceId <myResourceId,myResourceId,myResourceId>
```

To move the resources to different subscription, include the **-DestinationSubscriptionId** parameter.

```
Move-AzResource -DestinationSubscriptionId "<myDestinationSubscriptionID>" `  
-DestinationResourceGroupName "<myDestinationResourceGroup>" `  
-ResourceId <myResourceId,myResourceId,myResourceId>
```

When you are asked to confirm that you want to move the specified resources, enter Y to confirm.

## Next steps

You can move many different types of resources between resource groups and subscriptions. For more information, see [Move resources to a new resource group or subscription](#).

# Move a Marketplace Azure Virtual Machine to another subscription

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

To move a Marketplace virtual machine to a different subscription, you must move the OS disk to that subscription and then recreate the virtual machine.

You don't need this procedure to move a data disk to a new subscription. Instead, create a new virtual machine in the new subscription from the Marketplace, then move and attach the data disk.

This script demonstrates three operations:

- Create a snapshot of an OS disk.
- Move the snapshot to a different subscription.
- Create a virtual machine based on that snapshot.

## Prerequisites

You can use either the Azure Cloud Shell or a local Azure CLI.

- [Azure Cloud Shell](#) with the Bash environment. Or launch the Cloud Shell here.

 [Launch Cloud Shell](#)

- Local Azure CLI, see how to [install the Azure CLI](#). If you're running on Windows or macOS, consider running Azure CLI in a Docker container. For more information, see [How to run the Azure CLI in a Docker container](#).
  - Sign in to the Azure CLI by using the `az login` command. To finish the authentication process, follow the steps displayed in your terminal. For other sign-in options, see [Sign in with the Azure CLI](#).
  - When you first use Azure CLI, install the Azure CLI extension. For more information about extensions, see [Use extensions with the Azure CLI](#).
  - Run `az version` to find the version and dependent libraries that are installed. To upgrade to the latest version, run `az upgrade`.

## Sample script

To move a Marketplace virtual machine to a different subscription, you must create a new virtual machine for the same Marketplace offer from the moved OS disk.

### NOTE

If the virtual machine plan is no longer available in the Marketplace, you can't use this procedure.

```
#!/bin/bash
# Set variable values before proceeding.
```

```

# Variables
sourceResourceGroup= Resource group for the current virtual machine
sourceSubscription= Subscription for the current virtual machine
vmName= Name of the current virtual machine

destinationResourceGroup= Resource group for the new virtual machine, create if necessary
destinationSubscription= Subscription for the new virtual machine

# Set your current subscription for the source virtual machine
az account set --subscription $sourceSubscription

# Load variables about your virtual machine
# osType = windows or linux
osType=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--name $vmName --subscription $sourceSubscription \
--query 'storageProfile.osDisk.osType' --output tsv)

# offer = Your offer in Marketplace
offer=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--name $vmName --query 'storageProfile.imageReference.offer' --output tsv)

# plan = Your plan in Marketplace
plan=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--name $vmName --query 'plan' --output tsv)

# publisher = Your publisher in Marketplace
publisher=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--name $vmName --query 'storageProfile.imageReference.publisher' --output tsv)

# Get information to create new virtual machine
planName=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--subscription $sourceSubscription --query 'plan.name' --name $vmName)
planProduct=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--subscription $sourceSubscription --query 'plan.product' --name $vmName)
planPublisher=$(az vm get-instance-view --resource-group $sourceResourceGroup \
--subscription $sourceSubscription --query 'plan.publisher' --name $vmName)

# Get the name of the OS disk
osDiskName=$(az vm show --resource-group $sourceResourceGroup --name $vmName \
--query 'storageProfile.osDisk.name' --output tsv)

# Verify the terms for your market virtual machine
az vm image terms show --offer $offer --plan '$plan' --publisher $publisher \
--subscription $sourceSubscription

# Deallocate the virtual machine
az vm deallocate --resource-group $sourceResourceGroup --name $vmName

# Create a snapshot of the OS disk
az snapshot create --resource-group $sourceResourceGroup --name MigrationSnapshot \
--source
"/subscriptions/$sourceSubscription/resourceGroups/$sourceResourceGroup/providers/Microsoft.Compute/disks/$o
sDiskName"

# Move the snapshot to your destination resource group
az resource move --destination-group $destinationResourceGroup \
--destination-subscription-id $destinationSubscription \
--ids
"/subscriptions/$sourceSubscription/resourceGroups/$sourceResourceGroup/providers/Microsoft.Compute/snapshot
s/MigrationSnapshot"

# Set your subscription to the destination value
az account set --subscription $destinationSubscription

# Accept the terms from the Marketplace
az vm image terms accept --offer $offer --plan '$plan' --publisher $publisher \
--subscription $destinationSubscription

# Create disk from the snapshot

```

```
az disk create --resource-group $destinationResourceGroup --name DestinationDisk \
    --source
"/subscriptions/$destinationSubscription/resourceGroups/$destinationResourceGroup/providers/Microsoft.Compute/snapshots/MigrationSnapshot" \
    --os-type $osType

# Create virtual machine from disk
az vm create --resource-group $destinationResourceGroup --name $vmName \
    --plan-name $planName --plan-product $planProduct --plan-publisher $planPublisher \
    --attach-os-disk
"/subscriptions/$destinationSubscription/resourceGroups/$destinationResourceGroup/providers/Microsoft.Compute/disks/DestinationDisk" \
    --os-type $osType
```

## Clean up resources

After the sample has been run, use the following commands to remove the resource groups and all associated resources:

```
az group delete --name $sourceResourceGroup --subscription $sourceSubscription
az group delete --name $destinationResourceGroup --subscription $destinationSubscription
```

## Azure CLI references used in this article

- [az account set](#)
- [az disk create](#)
- [az group delete](#)
- [az resource move](#)
- [az snapshot create](#)
- [az vm create](#)
- [az vm deallocate](#)
- [az vm delete](#)
- [az vm get-instance-view](#)
- [az vm image terms accept](#)
- [az vm image terms show](#)
- [az vm show](#)

## Next steps

- [Move VMs to another Azure region](#)
- [Move a VM to another subscription or resource group](#)

# Move VMs to another Azure region

9/21/2022 • 7 minutes to read • [Edit Online](#)

There are scenarios in which you'd want to move your existing Azure IaaS virtual machines (VMs) from one region to another. For example, you want to improve reliability and availability of your existing VMs, to improve manageability, or to move for governance reasons. For more information, see the [Azure VM move overview](#).

You can use [Azure Site Recovery](#) service to move Azure VMs to a secondary region.

In this tutorial, you learn how to:

- Verify prerequisites for the move
- Prepare the source VMs and the target region
- Copy the data and enable replication
- Test the configuration and perform the move
- Delete the resources in the source region

## IMPORTANT

To move Azure VMs to another region, we now recommend using [Azure Resource Mover](#). Resource Mover is in public preview and provides:

- A single hub for moving resources across regions.
- Reduced move time and complexity. Everything you need is in a single location.
- A simple and consistent experience for moving different types of Azure resources.
- An easy way to identify dependencies across resources you want to move. This helps you to move related resources together, so that everything works as expected in the target region, after the move.
- Automatic cleanup of resources in the source region, if you want to delete them after the move.
- Testing. You can try out a move, and then discard it if you don't want to do a full move.

## NOTE

This tutorial shows you how to move Azure VMs from one region to another as is. If you need to improve availability by moving VMs in an availability set to zone pinned VMs in a different region, see the [Move Azure VMs into Availability Zones tutorial](#).

## Prerequisites

- Make sure that the Azure VMs are in the Azure region from which you want to move.
- Verify that your choice of [source region - target region combination is supported](#), and make an informed decision about the target region.
- Make sure that you understand the [scenario architecture and components](#).
- Review the [support limitations and requirements](#).
- Verify account permissions. If you created your free Azure account, you're the administrator of your subscription. If you're not the subscription administrator, work with the administrator to assign the permissions that you need. To enable replication for a VM and essentially copy data by using Azure Site Recovery, you must have:

- Permissions to create a VM in Azure resources. The Virtual Machine Contributor built-in role has these permissions, which include:
  - Permission to create a VM in the selected resource group
  - Permission to create a VM in the selected virtual network
  - Permission to write to the selected storage account
- Permissions to manage Azure Site Recovery operations. The Site Recovery Contributor role has all the permissions that are required to manage Site Recovery operations in a Recovery Services vault.
- Make sure that all the latest root certificates are on the Azure VMs that you want to move. If the latest root certificates aren't on the VM, security constraints will prevent the data copy to the target region.
- For Windows VMs, install all the latest Windows updates on the VM, so that all the trusted root certificates are on the machine. In a disconnected environment, follow the standard Windows Update and certificate update processes for your organization.
- For Linux VMs, follow the guidance provided by your Linux distributor to get the latest trusted root certificates and certificate revocation list on the VM.
- Make sure that you're not using an authentication proxy to control network connectivity for VMs that you want to move.
- If the VM that you're trying to move doesn't have access to the internet, or it's using a firewall proxy to control outbound access, [check the requirements](#).
- Identify the source networking layout and all the resources that you're currently using. This includes but isn't limited to load balancers, network security groups (NSGs), and public IPs.
- Verify that your Azure subscription allows you to create VMs in the target region that's used for disaster recovery. Contact support to enable the required quota.
- Make sure that your subscription has enough resources to support VMs with sizes that match your source VMs. If you're using Site Recovery to copy data to the target, Site Recovery chooses the same size or the closest possible size for the target VM.
- Make sure that you create a target resource for every component that's identified in the source networking layout. This step is important to ensure that your VMs have all the functionality and features in the target region that you had in the source region.

**NOTE**

Azure Site Recovery automatically discovers and creates a virtual network when you enable replication for the source VM. You can also pre-create a network and assign it to the VM in the user flow for enable replication. As mentioned later, you need to manually create any other resources in the target region.

To create the most commonly used network resources that are relevant for you based on the source VM configuration, see the following documentation:

- [Network security groups](#)
- [Load balancers](#)
- [Public IP](#)
- For any other networking components, see the [networking documentation](#).

## Prepare

The following steps shows how to prepare the virtual machine for the move using Azure Site Recovery as a solution.

### Create the vault in any region, except the source region

1. Sign in to the [Azure portal](#)
2. In search, type Recovery Services > click Recovery Services vaults
3. In Recovery Services vaults menu, click +Add.
4. In **Name**, specify the friendly name **ContosoVMVault**. If you have more than one subscription, select the appropriate one.
5. Create the resource group **ContosoRG**.
6. Specify an Azure region. To check supported regions, see geographic availability in [Azure Site Recovery pricing details](#).
7. In **Recovery Services vaults**, select **ContosoVMVault** > **Replicated items** > +Replicate.
8. In the dropdown, select **Azure Virtual Machines**.
9. In **Source location**, select the source Azure region where your VMs are currently running.
10. Select the Resource Manager deployment model. Then select the **Source subscription** and **Source resource group**.
11. Select **OK** to save the settings.

### Enable replication for Azure VMs and start copying the data

Site Recovery retrieves a list of the VMs that are associated with the subscription and resource group.

1. In the next step, select the VM that you want to move, then select **OK**.
2. In **Settings**, select **Disaster recovery**.
3. In **Configure disaster recovery** > **Target region**, select the target region to which you'll replicate.
4. For this tutorial, accept the other default settings.
5. Select **Enable replication**. This step starts a job to enable replication for the VM.

## Move

The following steps shows how to perform the move to the target region.

1. Go to the vault. In **Settings** > **Replicated items**, select the VM, and then select **Failover**.
2. In **Failover**, select **Latest**.
3. Select **Shut down machine before beginning failover**. Site Recovery attempts to shut down the source VM before triggering the failover. Failover continues even if shutdown fails. You can follow the failover progress on the **Jobs** page.
4. After the job is finished, check that the VM appears in the target Azure region as expected.

## Discard

In case you checked the moved VM and need to make changes to point of failover or want to go back to a previous point, in the **Replicated items**, right-select the VM > **Change recovery point**. This step provides you the option to specify a different recovery point and failover to that one.

## Commit

Once you have checked the moved VM and are ready to commit the change, in the **Replicated items**, right-select the VM > **Commit**. This step finishes the move process to the target region. Wait until the commit job finishes.

## Clean up

The following steps will guide you through how to clean up the source region as well as related resources that were used for the move.

For all resources that were used for the move:

- Go to the VM. Select **Disable Replication**. This step stops the process from copying the data for the VM.

### IMPORTANT

It's important to perform this step to avoid being charged for Azure Site Recovery replication.

If you have no plans to reuse any of the source resources, complete these additional steps:

1. Delete all the relevant network resources in the source region that you identified in [prerequisites](#).
2. Delete the corresponding storage account in the source region.

## Next steps

In this tutorial, you moved an Azure VM to a different Azure region. Now you can configure disaster recovery for the VM that you moved.

[Set up disaster recovery after migration](#)

# Move Azure VMs into Availability Zones

9/21/2022 • 8 minutes to read • [Edit Online](#)

This article describes how to move Azure VMs to an availability zone in a different region. If you want to move to a different zone in the same region, [review this article](#).

Availability Zones in Azure help protect your applications and data from datacenter failures. Each Availability Zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region helps protect applications and data from datacenter failures. With Availability Zones, Azure offers a service-level agreement (SLA) of 99.99% for uptime of virtual machines (VMs). Availability Zones are supported in select regions, as mentioned in [Regions that support Availability Zones](#).

In a scenario where your VMs are deployed as *single instance* into a specific region, and you want to improve your availability by moving these VMs into an Availability Zone, you can do so by using Azure Site Recovery. This action can further be categorized into:

- Move single-instance VMs into Availability Zones in a target region
- Move VMs in an availability set into Availability Zones in a target region

## IMPORTANT

To move Azure VMs to an availability zone in a different region, we now recommend using [Azure Resource Mover](#). Resource Mover is in public preview and provides:

- A single hub for moving resources across regions.
- Reduced move time and complexity. Everything you need is in a single location.
- A simple and consistent experience for moving different types of Azure resources.
- An easy way to identify dependencies across resources you want to move. This helps you to move related resources together, so that everything works as expected in the target region, after the move.
- Automatic cleanup of resources in the source region, if you want to delete them after the move.
- Testing. You can try out a move, and then discard it if you don't want to do a full move.

## Check prerequisites

- Check whether the target region has [support for Availability Zones](#). Check that your choice of [source region/target region combination is supported](#). Make an informed decision on the target region.
- Make sure that you understand the [scenario architecture and components](#).
- Review the [support limitations and requirements](#).
- Check account permissions. If you just created your free Azure account, you're the admin of your subscription. If you aren't the subscription admin, work with the admin to assign the permissions you need. To enable replication for a VM and eventually copy data to the target by using Azure Site Recovery, you must have:
  1. Permission to create a VM in Azure resources. The *Virtual Machine Contributor* built-in role has these permissions, which include:
    - Permission to create a VM in the selected resource group
    - Permission to create a VM in the selected virtual network

- Permission to write to the selected storage account
2. Permission to manage Azure Site Recovery tasks. The *Site Recovery Contributor* role has all permissions required to manage Site Recovery actions in a Recovery Services vault.

## Prepare the source VMs

1. Your VMs should use managed disks if you want to move them to an Availability Zone by using Site Recovery. You can convert existing Windows VMs that use unmanaged disks to use managed disks. Follow the steps at [Convert a Windows virtual machine from unmanaged disks to managed disks](#). Ensure that the availability set is configured as *managed*.
2. Check that all the latest root certificates are present on the Azure VMs you want to move. If the latest root certificates aren't present, the data copy to the target region can't be enabled because of security constraints.
3. For Windows VMs, install all the latest Windows updates on the VM, so that all the trusted root certificates are on the machine. In a disconnected environment, follow the standard Windows update and certificate update processes for your organization.
4. For Linux VMs, follow the guidance provided by your Linux distributor to get the latest trusted root certificates and certificate revocation list on the VM.
5. Make sure you don't use an authentication proxy to control network connectivity for VMs that you want to move.
6. Verify [outbound connectivity requirements for VMs](#).
7. Identify the source networking layout and the resources you currently use for verification, including load balancers, NSGs, and public IP.

## Prepare the target region

1. Check that your Azure subscription lets you create VMs in the target region used for disaster recovery. If necessary, contact support to enable the required quota.
2. Make sure your subscription has enough resources to support VMs with sizes that match your source VMs. If you use Site Recovery to copy data to the target, it picks the same size or the closest possible size for the target VM.
3. Create a target resource for every component identified in the source networking layout. This action ensures that after you cut over to the target region, your VMs have all the functionality and features that you had in the source.

### NOTE

Azure Site Recovery automatically discovers and creates a virtual network and storage account when you enable replication for the source VM. You can also pre-create these resources and assign to the VM as part of the enable replication step. But for any other resources, as mentioned later, you need to manually create them in the target region.

The following documents tell how to create the most commonly used network resources that are relevant to you, based on the source VM configuration.

- [Network security groups](#)
- [Load balancers](#)
- [Public IP](#)

For any other networking components, refer to the networking [documentation](#).

**IMPORTANT**

Ensure that you use a zone-redundant load balancer in the target. You can read more at [Standard Load Balancer and Availability Zones](#).

4. Manually [create a non-production network](#) in the target region if you want to test the configuration before you cut over to the target region. We recommend this approach because it causes minimal interference with the production environment.

## Enable replication

The following steps will guide you when using Azure Site Recovery to enable replication of data to the target region, before you eventually move them into Availability Zones.

**NOTE**

These steps are for a single VM. You can extend the same to multiple VMs. Go to the Recovery Services vault, select + **Replicate**, and select the relevant VMs together.

1. In the Azure portal, select **Virtual machines**, and select the VM you want to move into Availability Zones.
2. In **Operations**, select **Disaster recovery**.
3. In **Configure disaster recovery > Target region**, select the target region to which you'll replicate. Ensure this region [supports](#) Availability Zones.
4. Select **Next: Advanced settings**.
5. Choose the appropriate values for the target subscription, target VM resource group, and virtual network.
6. In the **Availability** section, choose the Availability Zone into which you want to move the VM.

**NOTE**

If you don't see the option for availability set or Availability Zone, ensure that the [prerequisites](#) are met and the [preparation](#) of source VMs is complete.

7. Select **Enable Replication**. This action starts a job to enable replication for the VM.

## Check settings

After the replication job has finished, you can check the replication status, modify replication settings, and test the deployment.

1. In the VM menu, select **Disaster recovery**.
2. You can check replication health, the recovery points that have been created and the source, and target regions on the map.

## Test the configuration

1. In the virtual machine menu, select **Disaster recovery**.

2. Select the **Test Failover** icon.
3. In **Test Failover**, select a recovery point to use for the failover:
  - **Latest processed**: Fails the VM over to the latest recovery point that was processed by the Site Recovery service. The time stamp is shown. With this option, no time is spent processing data, so it provides a low recovery time objective (RTO).
  - **Latest app-consistent**: This option fails over all VMs to the latest app-consistent recovery point. The time stamp is shown.
  - **Custom**: Select any recovery point.
4. Select the test target Azure virtual network to which you want to move the Azure VMs to test the configuration.

**IMPORTANT**

We recommend that you use a separate Azure VM network for the test failure, and not the production network in the target region into which you want to move your VMs.

5. To start testing the move, select **OK**. To track progress, select the VM to open its properties. Or, you can select the **Test Failover** job in the vault name > **Settings** > **Jobs** > **Site Recovery jobs**.
6. After the failover finishes, the replica Azure VM appears in the Azure portal > **Virtual Machines**. Make sure that the VM is running, sized appropriately, and connected to the appropriate network.
7. If you want to delete the VM created as part of testing the move, select **Cleanup test failover** on the replicated item. In **Notes**, record and save any observations associated with the test.

## Move to the target region and confirm

1. In the virtual machine menu, select **Disaster recovery**.
2. Select the **Failover** icon.
3. In **Failover**, select **Latest**.
4. Select **Shut down machine before beginning failover**. Site Recovery attempts to shut down the source VM before triggering the failover. Failover continues even if shutdown fails. You can follow the failover progress on the **Jobs** page.
5. After the job is finished, check that the VM appears in the target Azure region as expected.
6. In **Replicated items**, right-click the VM > **Commit**. This finishes the move process to the target region. Wait until the commit job is finished.

## Discard the resource in the source region

Go to the VM. Select **Disable Replication**. This action stops the process of copying the data for the VM.

**IMPORTANT**

Do the preceding step to avoid getting charged for Site Recovery replication after the move. The source replication settings are cleaned up automatically. Note that the Site Recovery extension that is installed as part of the replication isn't removed and needs to be removed manually.

## Next steps

In this tutorial, you increased the availability of an Azure VM by moving into an availability set or Availability Zone. Now you can set disaster recovery for the moved VM.

Set up disaster recovery after migration

# Move a Maintenance Control configuration to another region

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Follow this article to move a Maintenance Control configuration to a different Azure region. You might want to move a configuration for a number of reasons. For example, to take advantage of a new region, to deploy features or services available in a specific region, to meet internal policy and governance requirements, or in response to capacity planning.

[Maintenance control](#), with customized maintenance configurations, allows you to control how platform updates are applied to VMs, and to Azure Dedicated Hosts. There are a couple of scenarios for moving maintenance control across regions:

- To move your maintenance control configuration, but not the resources associated with the configuration, follow the instructions in this article.
- To move the resources associated with a maintenance configuration, but not the configuration itself, follow [these instructions](#).
- To move both the maintenance configuration and the resources associated with it, first follow the instructions in this article. Then, follow [these instructions](#).

## Prerequisites

Before you begin moving a maintenance control configuration:

- Maintenance configurations are associated with Azure VMs or Azure Dedicated Hosts. Make sure that VM/host resources exist in the new region before you begin.
- Identify:
  - Existing maintenance control configurations.
  - The resource groups in which existing configurations currently reside.
  - The resource groups to which the configurations will be added after moving to the new region.
  - The resources associated with the maintenance configuration you want to move.
  - Check that the resources in the new region are the same as those associated with the current maintenance configurations. The configurations can have the same names in the new region as they did in the old, but this isn't required.

## Prepare and move

1. Retrieve all of the maintenance configurations in each subscription. Run the CLI [az maintenance configuration list](#) command to do this, replacing \$subId with your subscription ID.

```
az maintenance configuration list --subscription $subId --query "[*].{Name:name, Location:location, ResGroup:resourceGroup}" --output table
```

2. Review the returned table list of configuration records within the subscription. Here's an example. Your list will contain values for your specific environment.

NAME	LOCATION	RESOURCE GROUP
Skip Maintenance	eastus2	configuration-resource-group
IgniteDemoConfig	eastus2	configuration-resource-group
defaultMaintenanceConfiguration-eastus	eastus	test-configuration

3. Save your list for reference. As you move the configurations, it helps you to verify that everything's been moved.
4. As a reference, map each configuration/resource group to the new resource group in the new region.
5. Create new maintenance configurations in the new region using [PowerShell](#), or [CLI](#).
6. Associate the configurations with the resources in the new region, using [PowerShell](#), or [CLI](#).

## Verify the move

After moving the configurations, compare configurations and resources in the new region with the table list you prepared.

## Clean up source resources

After the move, consider deleting the moved maintenance configurations in the source region, [PowerShell](#), or [CLI](#).

## Next steps

Follow [these instructions](#) if you need to move resources associated with maintenance configurations.

# Move resources in a Maintenance Control configuration to another region

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

Follow this article to move resources associated with a Maintenance Control configuration to a different Azure region. You might want to move a configuration for a number of reasons. For example, to take advantage of a new region, to deploy features or services available in a specific region, to meet internal policy and governance requirements, or in response to capacity planning.

[Maintenance control](#), with customized maintenance configurations, allows you to control how platform updates are applied to VMs, and to Azure Dedicated Hosts. There are a couple of scenarios for moving maintenance control across regions:

- To move the resources associated with a maintenance configuration, but not the configuration itself, follow this article.
- To move your maintenance control configuration, but not the resources associated with the configuration, follow [these instructions](#).
- To move both the maintenance configuration and the resources associated with it, first follow [these instructions](#). Then, follow the instructions in this article.

## Prerequisites

Before you begin moving the resources associated with a Maintenance Control configuration:

- Make sure that the resources you're moving exist in the new region before you begin.
- Verify the Maintenance Control configurations associated with the Azure VMs and Azure Dedicated Hosts that you want to move. Check each resource individually. There's currently no way to retrieve configurations for multiple resources.
- When retrieving configurations for a resource:
  - Make sure you use the subscription ID for the account, not an Azure Dedicated Host ID.
  - CLI: The --output table parameter is used for readability only, and can be deleted or changed.
  - PowerShell: The Format-Table Name parameter is used for readability only, and can be deleted or changed.
  - If you use PowerShell, you get an error if you try to list configurations for a resource that doesn't have any associated configurations. The error will be similar to: "Operation failed with status: 'Not Found'. Details: 404 Client Error: Not Found for url".

## Prepare to move

1. Before you start, define these variables. We've provided an example for each.

VARIABLE	DETAILS	EXAMPLE
\$subId	ID for subscription containing the maintenance configurations	"our-subscription-ID"
\$srcGroupName	Resource group name (Azure VM)	"VMResourceGroup"

VARIABLE	DETAILS	EXAMPLE
\$vmName	VM resource name	"myVM"
\$adhRsrcGroupName	Resource group (Dedicated hosts)	"HostResourceGroup"
\$adh	Dedicated host name	"myHost"
\$adhParentName	Parent resource name	"HostGroup"

2. To retrieve the maintenance configurations using the PowerShell [Get-AZConfigurationAssignment](#) command:

- For Azure Dedicated Hosts, run:

```
Get-AzConfigurationAssignment -ResourceGroupName $adhRsrcGroupName -ResourceName $adh -  
ResourceType hosts -ProviderName Microsoft.Compute -ResourceParentName $adhParentName -  
ResourceParentType hostGroups | Format-Table Name
```

- For Azure VMs, run:

```
Get-AzConfigurationAssignment -ResourceGroupName $rgName -ResourceName $vmName -ProviderName  
Microsoft.Compute -ResourceType virtualMachines | Format-Table Name
```

3. To retrieve the maintenance configurations using the CLI [az maintenance assignment](#) command:

- For Azure Dedicated Hosts:

```
az maintenance assignment list --subscription $subId --resource-group $adhRsrcGroupName --  
resource-name $adh --resource-type hosts --provider-name Microsoft.Compute --resource-parent-  
name $adhParentName --resource-parent-type hostGroups --query "[].  
{HostResourceGroup:resourceGroup,ConfigName:name}" --output table
```

- For Azure VMs:

```
az maintenance assignment list --subscription $subId --provider-name Microsoft.Compute --  
resource-group $rsrcGroupName --resource-name $vmName --resource-type virtualMachines --query  
"[].{HostResourceGroup:resourceGroup, ConfigName:name}" --output table
```

## Move

1. [Follow these instructions](#) to move the Azure VMs to the new region.
2. After the resources are moved, reapply maintenance configurations to the resources in the new region as appropriate, depending on whether you moved the maintenance configurations. You can apply a maintenance configuration to a resource using [PowerShell](#) or [CLI](#).

## Verify the move

Verify resources in the new region, and verify associated configurations for the resources in the new region.

## Clean up source resources

After the move, consider deleting the moved resources in the source region.

## Next steps

Follow [these instructions](#) if you need to move maintenance configurations.

# Migrate from Amazon Web Services (AWS) and other platforms to managed disks in Azure

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

You can upload VHD files from AWS or on-premises virtualization solutions to Azure to create virtual machines (VMs) that use managed disks. Azure managed disks removes the need to manage storage accounts for Azure IaaS VMs. You just specify the type of disk, and the size of that disk that you need, and Azure creates and manages the disk for you.

You can upload either generalized and specialized VHDs.

- **Generalized VHD** - has had all of your personal account information removed using Sysprep.
- **Specialized VHD** - maintains the user accounts, applications, and other state data from your original VM.

## IMPORTANT

Before uploading any VHD to Azure, you should follow [Prepare a Windows VHD or VHDX to upload to Azure](#)

SCENARIO	DOCUMENTATION
You have existing AWS EC2 instances that you would like to migrate to Azure VMs using managed disks	<a href="#">Move a VM from Amazon Web Services (AWS) to Azure</a>
You have a VM from another virtualization platform that you would like to use as an image to create multiple Azure VMs.	<a href="#">Upload a generalized VHD and use it to create a new VM in Azure</a>
You have a uniquely customized VM that you would like to recreate in Azure.	<a href="#">Upload a specialized VHD to Azure and create a new VM</a>

## Overview of managed disks

Azure managed disks simplifies VM management by removing the need to manage storage accounts. Managed disks also benefit from better reliability of VMs in an Availability Set. It ensures that the disks of different VMs in an Availability Set are sufficiently isolated from each other to avoid a single point of failure. It automatically places disks of different VMs in an Availability Set in different Storage scale units (stamps) which limits the impact of single Storage scale unit failures caused due to hardware and software failures. Based on your needs, you can choose from four types of storage options. To learn about the available disk types, see our article [Select a disk type](#).

## Plan for the migration to managed disks

This section helps you to make the best decision on VM and disk types.

If you are planning on migrating from unmanaged disks to managed disks, you should be aware that users with the [Virtual Machine Contributor](#) role will not be able to change the VM size (as they could pre-conversion). This is because VMs with managed disks require the user to have the Microsoft.Compute/disks/write permission on the OS disks.

## Location

Pick a location where Azure managed disks are available. If you are migrating to premium SSDs, also ensure that premium storage is available in the region where you are planning to migrate to. See [Azure Services by Region](#) for up-to-date information on available locations.

## VM sizes

If you are migrating to premium SSDs, you have to update the size of the VM to premium storage capable size available in the region where VM is located. Review the VM sizes that are premium storage capable. The Azure VM size specifications are listed in [Sizes for virtual machines](#). Review the performance characteristics of virtual machines that work with premium storage and choose the most appropriate VM size that best suits your workload. Make sure that there is sufficient bandwidth available on your VM to drive the disk traffic.

## Disk sizes

For information on available disk types and sizes, see [What disk types are available in Azure?](#).

## Disk caching policy

### Premium Managed Disks

By default, disk caching policy is *Read-Only* for all the Premium data disks, and *Read-Write* for the Premium operating system disk attached to the VM. This configuration setting is recommended to achieve the optimal performance for your application's IOs. For write-heavy or write-only data disks (such as SQL Server log files), disable disk caching so that you can achieve better application performance.

## Pricing

Review the [pricing for managed disks](#).

## Next Steps

- Before uploading any VHD to Azure, you should follow [Prepare a Windows VHD or VHDX to upload to Azure](#)

# Move a Windows VM from Amazon Web Services (AWS) to an Azure virtual machine

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

If you are evaluating Azure virtual machines for hosting your workloads, you can export an existing Amazon Web Services (AWS) EC2 Windows VM instance then upload the virtual hard disk (VHD) to Azure. Once the VHD is uploaded, you can create a new VM in Azure from the VHD.

This article covers moving a single VM from AWS to Azure. If you want to move VMs from AWS to Azure at scale, see [Migrate virtual machines in Amazon Web Services \(AWS\) to Azure with Azure Site Recovery](#).

## Prepare the VM

You can upload both generalized and specialized VHDs to Azure. Each type requires that you prepare the VM before exporting from AWS.

- **Generalized VHD** - a generalized VHD has had all of your personal account information removed using Sysprep. If you intend to use the VHD as an image to create new VMs from, you should:
  - [Prepare a Windows VM](#).
  - Generalize the virtual machine using Sysprep.
- **Specialized VHD** - a specialized VHD maintains the user accounts, applications and other state data from your original VM. If you intend to use the VHD as-is to create a new VM, ensure the following steps are completed.
  - [Prepare a Windows VHD to upload to Azure](#). **Do not** generalize the VM using Sysprep.
  - Remove any guest virtualization tools and agents that are installed on the VM (i.e. VMware tools).
  - Ensure the VM is configured to pull its IP address and DNS settings via DHCP. This ensures that the server obtains an IP address within the VNet when it starts up.

## Export and download the VHD

Export the EC2 instance to a VHD in an Amazon S3 bucket. Follow the steps in the Amazon documentation article [Exporting an Instance as a VM Using VM Import/Export](#) and run the `create-instance-export-task` command to export the EC2 instance to a VHD file.

The exported VHD file is saved in the Amazon S3 bucket you specify. The basic syntax for exporting the VHD is below, just replace the placeholder text in <brackets> with your information.

```
aws ec2 create-instance-export-task --instance-id <instanceID> --target-environment Microsoft \
--export-to-s3-task DiskImageFormat=VHD,ContainerFormat=ova,S3Bucket=<bucket>,S3Prefix=<prefix>
```

Once the VHD has been exported, follow the instructions in [How Do I Download an Object from an S3 Bucket?](#) to download the VHD file from the S3 bucket.

**IMPORTANT**

AWS charges data transfer fees for downloading the VHD. See [Amazon S3 Pricing](#) for more information.

## Next steps

Now you can upload the VHD to Azure and create a new VM.

- If you ran Sysprep on your source to **generalize** it before exporting, see [Upload a generalized VHD and use it to create a new VMs in Azure](#)
- If you did not run Sysprep before exporting, the VHD is considered **specialized**, see [Upload a specialized VHD to Azure and create a new VM](#)
- Secure the VM using the Defender for Servers plan, which is part of [Microsoft Defender for Cloud](#).

# Create a Linux VM from a custom disk with the Azure CLI

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

This article shows you how to upload a customized virtual hard disk (VHD), and how to copy an existing VHD in Azure. The newly created VHD is then used to create new Linux virtual machines (VMs). You can install and configure a Linux distro to your requirements and then use that VHD to create a new Azure virtual machine.

To create multiple VMs from your customized disk, first create an image from your VM or VHD. For more information, see [Create a custom image of an Azure VM by using the CLI](#).

You have two options to create a custom disk:

- Upload a VHD
- Copy an existing Azure VM

## Requirements

To complete the following steps, you'll need:

- A Linux virtual machine that has been prepared for use in Azure. The [Prepare the VM](#) section of this article covers how to find distro-specific information on installing the Azure Linux Agent (waagent), which is needed for you to connect to a VM with SSH.
- The VHD file from an existing [Azure-endorsed Linux distribution](#) (or see [information for non-endorsed distributions](#)) to a virtual disk in the VHD format. Multiple tools exist to create a VM and VHD:
  - Install and configure [QEMU](#) or [KVM](#), taking care to use VHD as your image format. If needed, you can [convert an image](#) with `qemu-img convert`.
  - You can also use Hyper-V [on Windows 10](#) or [on Windows Server 2012/2012 R2](#).

### NOTE

The newer VHDX format is not supported in Azure. When you create a VM, specify VHD as the format. If needed, you can convert VHDX disks to VHD with `qemu-img convert` or the [Convert-VHD](#) PowerShell cmdlet. Azure does not support uploading dynamic VHDs, so you'll need to convert such disks to static VHDs before uploading. You can use tools such as [Azure VHD Utilities for GO](#) to convert dynamic disks during the process of uploading them to Azure.

- Make sure that you have the latest [Azure CLI](#) installed and you are signed in to an Azure account with `az login`.

In the following examples, replace example parameter names with your own values, such as `myResourceGroup`, `mystorageaccount`, and `mydisks`.

## Prepare the VM

Azure supports various Linux distributions (see [Endorsed Distributions](#)). The following articles describe how to prepare the various Linux distributions that are supported on Azure:

- [CentOS-based Distributions](#)

- [Debian Linux](#)
- [Oracle Linux](#)
- [Red Hat Enterprise Linux](#)
- [SLES & openSUSE](#)
- [Ubuntu](#)
- [Others: Non-Endorsed Distributions](#)

Also see the [Linux Installation Notes](#) for more general tips on preparing Linux images for Azure.

#### **NOTE**

The [Azure platform SLA](#) applies to VMs running Linux only when one of the endorsed distributions is used with the configuration details as specified under "Supported Versions" in [Linux on Azure-Endorsed Distributions](#).

## Option 1: Upload a VHD

You can now upload VHD straight into a managed disk. For instructions, see [Upload a VHD to Azure using Azure CLI](#).

## Option 2: Copy an existing VM

You can also create a customized VM in Azure and then copy the OS disk and attach it to a new VM to create another copy. This is fine for testing, but if you want to use an existing Azure VM as the model for multiple new VMs, create an *image* instead. For more information about creating an image from an existing Azure VM, see [Create a custom image of an Azure VM by using the CLI](#).

If you want to copy an existing VM to another region, you might want to use azcopy to [create a copy of a disk in another region](#).

Otherwise, you should take a snapshot of the VM and then create a new OS VHD from the snapshot.

### Create a snapshot

This example creates a snapshot of a VM named *myVM* in resource group *myResourceGroup* and creates a snapshot named *osDiskSnapshot*.

```
osDiskId=$(az vm show -g myResourceGroup -n myVM --query "storageProfile.osDisk.managedDisk.id" -o tsv)
az snapshot create \
    -g myResourceGroup \
    --source "$osDiskId" \
    --name osDiskSnapshot
```

### Create the managed disk

Create a new managed disk from the snapshot.

Get the ID of the snapshot. In this example, the snapshot is named *osDiskSnapshot* and it is in the *myResourceGroup* resource group.

```
snapshotId=$(az snapshot show --name osDiskSnapshot --resource-group myResourceGroup --query [id] -o tsv)
```

Create the managed disk. In this example, we will create a managed disk named *myManagedDisk* from our snapshot, where the disk is in standard storage and sized at 128 GB.

```
az disk create \
    --resource-group myResourceGroup \
    --name myManagedDisk \
    --sku Standard_LRS \
    --size-gb 128 \
    --source $snapshotId
```

## Create the VM

Create your VM with [az vm create](#) and attach (--attach-os-disk) the managed disk as the OS disk. The following example creates a VM named *myNewVM* using the managed disk you created from your uploaded VHD:

```
az vm create \
    --resource-group myResourceGroup \
    --location eastus \
    --name myNewVM \
    --os-type linux \
    --attach-os-disk myManagedDisk
```

You should be able to SSH into the VM with the credentials from the source VM.

## Next steps

After you have prepared and uploaded your custom virtual disk, you can read more about [using Resource Manager and templates](#). You may also want to [add a data disk](#) to your new VMs. If you have applications running on your VMs that you need to access, be sure to [open ports and endpoints](#).

# Migrating to Azure

9/21/2022 • 2 minutes to read • [Edit Online](#)

For migration, we recommend that you use the Azure Migrate service to migrate VMs and servers to Azure, rather than the Azure Site Recovery service. [Learn more](#) about Azure Migrate.

## Why use Azure Migrate?

Using Azure Migrate for migration provides a number of advantages:

- Azure Migrate provides a centralized hub for discovery, assessment, and migration to Azure.
- Using Azure Migrate provides interoperability and future extensibility with Azure Migrate tools, other Azure services, and third-party tools.
- The Azure Migrate:Server Migration tool is purpose-built for server migration to Azure. It's optimized for migration. You don't need to learn about concepts and scenarios that aren't directly relevant to migration.
- There are no tool usage charges for migration for 180 days, from the time replication is started for a VM. This gives you time to complete migration. You only pay for the storage and network resources used in replication, and for compute charges consumed during test migrations.
- Azure Migrate supports all migration scenarios supported by Site Recovery. In addition, for VMware VMs, Azure Migrate provides an agentless migration option.
- We're prioritizing new migration features for the Azure Migrate:Server Migration tool only. These features aren't targeted for Site Recovery.

## When to use Site Recovery?

Site Recovery should be used:

- For disaster recovery of on-premises machines to Azure.
- For disaster recovery of Azure VMs, between Azure regions.

Although we recommend using Azure Migrate to migrate on-premises servers to Azure, if you've already started your migration journey with Site Recovery, you can continue using it to complete your migration.

## Next steps

[Review common questions](#) about Azure Migrate.

# Use infrastructure automation tools with virtual machines in Azure

9/21/2022 • 7 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets ✓ Uniform scale sets

To create and manage Azure virtual machines (VMs) in a consistent manner at scale, some form of automation is typically desired. There are many tools and solutions that allow you to automate the complete Azure infrastructure deployment and management lifecycle. This article introduces some of the infrastructure automation tools that you can use in Azure. These tools commonly fit in to one of the following approaches:

- Automate the configuration of VMs
  - Tools include [Ansible](#), [Chef](#), [Puppet](#), and [Azure Resource Manager template](#).
  - Tools specific to VM customization include [cloud-init](#) for Linux VMs, [PowerShell Desired State Configuration \(DSC\)](#), and the [Azure Custom Script Extension](#) for all Azure VMs.
- Automate infrastructure management
  - Tools include [Packer](#) to automate custom VM image builds, and [Terraform](#) to automate the infrastructure build process.
  - [Azure Automation](#) can perform actions across your Azure and on-premises infrastructure.
- Automate application deployment and delivery
  - Examples include [Azure DevOps Services](#) and [Jenkins](#).

## Ansible

[Ansible](#) is an automation engine for configuration management, VM creation, or application deployment. Ansible uses an agent-less model, typically with SSH keys, to authenticate and manage target machines. Configuration tasks are defined in playbooks, with a number of Ansible modules available to carry out specific tasks. For more information, see [How Ansible works](#).

Learn how to:

- [Install and configure Ansible on Linux for use with Azure](#).
- [Create a Linux virtual machine](#).
- [Manage a Linux virtual machine](#).

## Chef

[Chef](#) is an automation platform that helps define how your infrastructure is configured, deployed, and managed. Additional components included Chef Habitat for application lifecycle automation rather than the infrastructure, and Chef InSpec that helps automate compliance with security and policy requirements. Chef Clients are installed on target machines, with one or more central Chef Servers that store and manage the configurations. For more information, see [An Overview of Chef](#).

Learn how to:

- [Deploy Chef Automate from the Azure Marketplace](#).
- [Install Chef on Windows and create Azure VMs](#).

## Puppet

Puppet is an enterprise-ready automation platform that handles the application delivery and deployment process. Agents are installed on target machines to allow Puppet Master to run manifests that define the desired configuration of the Azure infrastructure and VMs. Puppet can integrate with other solutions such as Jenkins and GitHub for an improved devops workflow. For more information, see [How Puppet works](#).

Learn how to:

- [Deploy Puppet](#).

## Cloud-init

Cloud-init is a widely used approach to customize a Linux VM as it boots for the first time. You can use cloud-init to install packages and write files, or to configure users and security. Because cloud-init is called during the initial boot process, there are no additional steps or required agents to apply your configuration. For more information on how to properly format your `#cloud-config` files, see the [cloud-init documentation site](#).

`#cloud-config` files are text files encoded in base64.

Cloud-init also works across distributions. For example, you don't use `apt-get install` or `yum install` to install a package. Instead you can define a list of packages to install. Cloud-init automatically uses the native package management tool for the distro you select.

We are actively working with our endorsed Linux distro partners in order to have cloud-init enabled images available in the Azure Marketplace. These images make your cloud-init deployments and configurations work seamlessly with VMs and virtual machine scale sets. Learn more details about cloud-init on Azure:

- [Cloud-init support for Linux virtual machines in Azure](#)
- [Try a tutorial on automated VM configuration using cloud-init](#).

## PowerShell DSC

PowerShell Desired State Configuration (DSC) is a management platform to define the configuration of target machines. DSC can also be used on Linux through the [Open Management Infrastructure \(OMI\) server](#).

DSC configurations define what to install on a machine and how to configure the host. A Local Configuration Manager (LCM) engine runs on each target node that processes requested actions based on pushed configurations. A pull server is a web service that runs on a central host to store the DSC configurations and associated resources. The pull server communicates with the LCM engine on each target host to provide the required configurations and report on compliance.

Learn how to:

- [Create a basic DSC configuration](#).
- [Configure a DSC pull server](#).
- [Use DSC for Linux](#).

## Azure Custom Script Extension

The Azure Custom Script Extension for [Linux](#) or [Windows](#) downloads and executes scripts on Azure VMs. You can use the extension when you create a VM, or any time after the VM is in use.

Scripts can be downloaded from Azure storage or any public location such as a GitHub repository. With the Custom Script Extension, you can write scripts in any language that runs on the source VM. These scripts can be used to install applications or configure the VM as desired. To secure credentials, sensitive information such as passwords can be stored in a protected configuration. These credentials are only decrypted inside the VM.

Learn how to:

- [Create a Linux VM with the Azure CLI and use the Custom Script Extension.](#)
- [Create a Windows VM with Azure PowerShell and use the Custom Script Extension.](#)

## Packer

[Packer](#) automates the build process when you create a custom VM image in Azure. You use Packer to define the OS and run post-configuration scripts that customize the VM for your specific needs. Once configured, the VM is then captured as a Managed Disk image. Packer automates the process to create the source VM, network and storage resources, run configuration scripts, and then create the VM image.

Learn how to:

- [Use Packer to create a Linux VM image in Azure.](#)
- [Use Packer to create a Windows VM image in Azure.](#)

## Terraform

[Terraform](#) is an automation tool that allows you to define and create an entire Azure infrastructure with a single template format language - the HashiCorp Configuration Language (HCL). With Terraform, you define templates that automate the process to create network, storage, and VM resources for a given application solution. You can use your existing Terraform templates for other platforms with Azure to ensure consistency and simplify the infrastructure deployment without needing to convert to an Azure Resource Manager template.

Learn how to:

- [Install and configure Terraform with Azure.](#)
- [Create an Azure infrastructure with Terraform.](#)

## Azure Automation

[Azure Automation](#) uses runbooks to process a set of tasks on the VMs you target. Azure Automation is used to manage existing VMs rather than to create an infrastructure. Azure Automation can run across both Linux and Windows VMs, as well as on-premises virtual or physical machines with a hybrid runbook worker. Runbooks can be stored in a source control repository, such as GitHub. These runbooks can then run manually or on a defined schedule.

Azure Automation also provides a Desired State Configuration (DSC) service that allows you to create definitions for how a given set of VMs should be configured. DSC then ensures that the required configuration is applied and the VM stays consistent. Azure Automation DSC runs on both Windows and Linux machines.

Learn how to:

- [Create a PowerShell runbook.](#)
- [Use Hybrid Runbook Worker to manage on-premises resources.](#)
- [Use Azure Automation DSC.](#)

## Azure DevOps Services

[Azure DevOps Services](#) is a suite of tools that help you share and track code, use automated builds, and create a complete continuous integration and development (CI/CD) pipeline. Azure DevOps Services integrates with Visual Studio and other editors to simplify usage. Azure DevOps Services can also create and configure Azure VMs and then deploy code to them.

Learn more about:

- [Azure DevOps Services](#).

## Jenkins

[Jenkins](#) is a continuous integration server that helps deploy and test applications, and create automated pipelines for code delivery. There are hundreds of plugins to extend the core Jenkins platform, and you can also integrate with many other products and solutions through webhooks. You can manually install Jenkins on an Azure VM, run Jenkins from within a Docker container, or use a pre-built Azure Marketplace image.

Learn how to:

- [Create a development infrastructure on a Linux VM in Azure with Jenkins, GitHub, and Docker](#).

## Azure Resource Manager template

[Azure Resource Manager](#) is the deployment and management service for Azure. It provides a management layer that enables you to create, update, and delete resources in your Azure subscription. You use management features, like access control, locks, and tags, to secure and organize your resources after deployment.

Learn how to:

- [Deploy Spot VMs using a Resource Manager template](#).
- [Create a Windows virtual machine from a Resource Manager template](#).
- [Download the template for a VM](#).
- [Create an Azure Image Builder template](#).

## Next steps

There are many different options to use infrastructure automation tools in Azure. You have the freedom to use the solution that best fits your needs and environment. To get started and try some of the tools built-in to Azure, see how to automate the customization of a [Linux](#) or [Windows](#) VM.

# User Data for Azure Virtual Machine

9/21/2022 • 3 minutes to read • [Edit Online](#)

User data allows you to pass your own scripts or metadata to your virtual machine.

## What is "user data"

User data is a set of scripts or other metadata, that will be inserted to an Azure virtual machine at provision time. Any application on the virtual machine can access the user data from the Azure Instance Metadata Service (IMDS) after provision.

User data is a new version of [custom data](#) and it offers added benefits:

- User data can be retrieved from Azure Instance Metadata Service(IMDS) after provision.
- User data is persistent. It will be available during the lifetime of the VM.
- User data can be updated from outside the VM, without stopping or rebooting the VM.
- User data can be queried via GET VM/VMSS API with \$expand option.

In addition, if user data is not added at provision time, you can still add it after provision.

### Security warning

#### WARNING

User data will not be encrypted, and any process on the VM can query this data. You should not store confidential information in user data.

Make sure you get the latest Azure Resource Manager API to use the new user data features. The contents should be base64 encoded before passed to the API. The size cannot exceed 64 KB.

## Create user data for Azure VM/VMSS

### Adding user data when creating new VM

Use [this Azure Resource Manager template](#) to create a new VM with user data. If you are using rest API, for single VMs, add 'UserData' to the "properties" section with the PUT request to create the VM.

```
{
  "name": "testVM",
  "location": "West US",
  "properties": {
    "hardwareProfile": {
      "vmSize": "Standard_A1"
    },
    "storageProfile": {
      "osDisk": {
        "osType": "Windows",
        "name": "osDisk",
        "createOption": "Attach",
        "vhd": {
          "uri": "http://myaccount.blob.core.windows.net/container/directory/blob.vhd"
        }
      }
    },
    "userData": "c2FtcGxlIHVzZXJEYXRh",
    "networkProfile": { "networkInterfaces" : [ { "name" : "nic1" } ] },
  }
}
```

## Adding user data when you create new virtual machine scale set

Using rest API, add 'UserData' to the "virtualMachineProfile" section with the PUT request when creating the virtual machine scale set.

```
{
  "location": "West US",
  "sku": {
    "name": "Standard_A1",
    "capacity": 1
  },
  "properties": {
    "upgradePolicy": {
      "mode": "Automatic"
    },
    "virtualMachineProfile": {
      "userData": "VXNlckRhdGE=",
      "osProfile": {
        "computerNamePrefix": "TestVM",
        "adminUsername": "TestUserName",
        "windowsConfiguration": {
          "provisionVMAgent": true,
          "timeZone": "Dateline Standard Time"
        }
      },
      "storageProfile": {
        "osDisk": {
          "createOption": "FromImage",
          "caching": "ReadOnly"
        },
        "imageReference": {
          "publisher": "publisher",
          "offer": "offer",
          "sku": "sku",
          "version": "1.2.3"
        }
      },
      "networkProfile": {"networkInterfaceConfigurations": [{"name": "nicconfig1", "properties": {"ipConfigurations": [{"name": "ip1", "properties": {"subnet": {"id": "vmssSubnet0"}}}]}]}},
      "diagnosticsProfile": {
        "bootDiagnostics": {
          "enabled": true,
          "storageUri": "https://crputest.blob.core.windows.net"
        }
      }
    },
    "provisioningState": 0,
    "overprovision": false,
    "uniqueId": "00000000-0000-0000-0000-000000000000"
  }
}
```

## Retrieving user data

Applications running inside the VM can retrieve user data from IMDS endpoint. For details, see [IMDS sample code here](#).

Customers can retrieve existing value of user data via rest API using \$expand=userData endpoint (request body can be left empty).

Single VMs:

```
GET "/subscriptions/{guid}/resourceGroups/{RGName}/providers/Microsoft.Compute/virtualMachines/{VMName}?
$expand=userData"
```

Virtual machine scale set:

```
GET "/subscriptions/{guid}/resourceGroups/{RGName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSSName}?
$expand=userData"
```

Virtual machine scale set VM:

```
GET  
"/subscriptions/{guid}/resourceGroups/{RGName}/providers/Microsoft.Compute/virtualMachineScaleSets/{VMSSName}/virtualmachines/  
instance id}?$expand=userData"
```

## Updating user data

With REST API, you can use a normal PUT or PATCH request to update the user data. The user data will be updated without the need to stop or reboot the VM.

```
PUT "/subscriptions/{guid}/resourceGroups/{RGName}/providers/Microsoft.Compute/ virtualMachines/{VMName}
```

```
PATCH "/subscriptions/{guid}/resourceGroups/{RGName}/providers/Microsoft.Compute/ virtualMachines/{VMName}
```

The VM.Properties in these requests should contain your desired UserData field, like this:

```
"properties": {  
    "hardwareProfile": {  
        "vmSize": "Standard_D1_v2"  
    },  
    "storageProfile": {  
        "imageReference": {  
            "sku": "2016-Datacenter",  
            "publisher": "MicrosoftWindowsServer",  
            "version": "latest",  
            "offer": "WindowsServer"  
        },  
        "osDisk": {  
            "caching": "ReadWrite",  
            "managedDisk": {  
                "storageAccountType": "StandardSSD_LRS"  
            },  
            "name": "vmOsdisk",  
            "createOption": "FromImage"  
        }  
    },  
    "networkProfile": {  
        "networkInterfaces": [  
            {  
                "id": "/subscriptions/{subscription-  
id}/resourceGroups/myResourceGroup/providers/Microsoft.Network/networkInterfaces/{existing-nic-name}",  
                "properties": {  
                    "primary": true  
                }  
            }  
        ]  
    },  
    "osProfile": {  
        "adminUsername": "{your-username}",  
        "computerName": "{vm-name}",  
        "adminPassword": "{your-password}"  
    },  
    "diagnosticsProfile": {  
        "bootDiagnostics": {  
            "storageUri": "http://{existing-storage-account-name}.blob.core.windows.net",  
            "enabled": true  
        }  
    },  
    "userData": "U29tZSBDDxN0b20gRGF0YQ=="  
}
```

### NOTE

If you pass in an empty string for "userData" in this case, the user data will be deleted.

## User data and custom data

Custom data will continue to work the same way as today. Note you cannot retrieve custom data from IMDS.

## Adding user data to an existing VM

If you have an existing VM/VMSS without user data, you can still add user data to this VM by using the updating commands, as described in the "[Updating the User data](#)" section. Make sure you upgrade to the latest version of Azure Resource Manager API.

## Next steps

Try out [Azure Instance Metadata Service](#), learn how to get the VM instance metadata and user data from its endpoint.

# Custom data and cloud-init on Azure Virtual Machines

9/21/2022 • 3 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

You might need to inject a script or other metadata into a Microsoft Azure virtual machine (VM) at provisioning time. In other clouds, this concept is often called *user data*. Microsoft Azure has a similar feature called *custom data*.

Custom data is made available to the VM during first startup or setup, which is called *provisioning*. Provisioning is the process where VM creation parameters (for example, host name, username, password, certificates, custom data, and keys) are made available to the VM. A provisioning agent, such as the [Linux Agent](#) or [cloud-init](#), processes those parameters.

## Pass custom data to the VM

To use custom data, you must Base64-encode the contents before passing the data to the API--unless you're using a CLI tool that does the conversion for you, such as the Azure CLI. The size can't exceed 64 KB.

In the CLI, you can pass your custom data as a file, as the following example shows. The file will be converted to Base64.

```
az vm create \
--resource-group myResourceGroup \
--name centos74 \
--image OpenLogic:CentOS-CI:7-CI:latest \
--custom-data cloud-init.txt \
--generate-ssh-keys
```

In Azure Resource Manager, there's a [base64 function](#):

```
"name": "[parameters('virtualMachineName')]",
"type": "Microsoft.Compute/virtualMachines",
"apiVersion": "2019-07-01",
"location": "[parameters('location')]",
"dependsOn": [
...],
"variables": {
    "customDataBase64": "[base64(parameters('stringData'))]"
},
"properties": {
...
    "osProfile": {
        "computerName": "[parameters('virtualMachineName')]",
        "adminUsername": "[parameters('adminUsername')]",
        "adminPassword": "[parameters('adminPassword')]",
        "customData": "[variables('customDataBase64')]"
    },
}
```

## Process custom data

The provisioning agents installed on the VMs handle communication with the platform and placing data on the

file system.

## Windows

Custom data is placed in `%SYSTEMDRIVE%\AzureData\CustomData.bin` as a binary file, but it isn't processed. If you want to process this file, you'll need to build a custom image and write code to process `CustomData.bin`.

## Linux

On Linux operating systems, custom data is passed to the VM via the `ovf-env.xml` file. That file is copied to the `/var/lib/waagent` directory during provisioning. Newer versions of the Linux Agent will also copy the Base64-encoded data to `/var/lib/waagent/CustomData` for convenience.

Azure currently supports two provisioning agents:

- **Linux Agent.** By default, the agent won't process custom data. You need to build a custom image with the data enabled. The [relevant settings](#) are:

- `Provisioning.DecodeCustomData`
- `Provisioning.ExecuteCustomData`

When you enable custom data and run a script, it will delay the VM reporting that it's ready or that provisioning has succeeded until the script has finished. If the script exceeds the total VM provisioning time allowance of 40 minutes, VM creation will fail.

If the script fails to run, or errors happen during execution, that's not a fatal provisioning failure. You'll need to create a notification path to alert you for the completion state of the script.

To troubleshoot custom data execution, review `/var/log/waagent.log`.

- **cloud-init.** By default, this agent will process custom data. It accepts [multiple formats](#) of custom data, such as cloud-init configuration and scripts.

Similar to the Linux Agent, if errors happen during execution of the configuration processing or scripts when cloud-init is processing the custom data, that's not a fatal provisioning failure. You'll need to create a notification path to alert you for the completion state of the script.

However, unlike the Linux Agent, cloud-init doesn't wait for custom data configurations from the user to finish before reporting to the platform that the VM is ready. For more information on cloud-init on Azure, including troubleshooting, see [cloud-init support for virtual machines in Azure](#).

## FAQ

### Can I update custom data after the VM has been created?

For single VMs, you can't update custom data in the VM model. But for virtual machine scale sets, you can update custom data via the [REST API](#), the [Azure CLI](#), or [Azure PowerShell](#). When you update custom data in the model for a virtual machine scale set:

- Existing instances in the scale set won't get the updated custom data until they're reimaged.
- Existing instances in the scale set that are upgraded won't get the updated custom data.
- New instances will receive the new custom data.

### Can I place sensitive values in custom data?

We advise *not* to store sensitive data in custom data. For more information, see [Azure data security and encryption best practices](#).

### Is custom data made available in IMDS?

Custom data is not available in Azure Instance Metadata Service (IMDS). We suggest using user data in IMDS instead. For more information, see [User data through Azure Instance Metadata Service](#).

# Configure the blue-green deployment strategy for Azure Linux virtual machines

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs

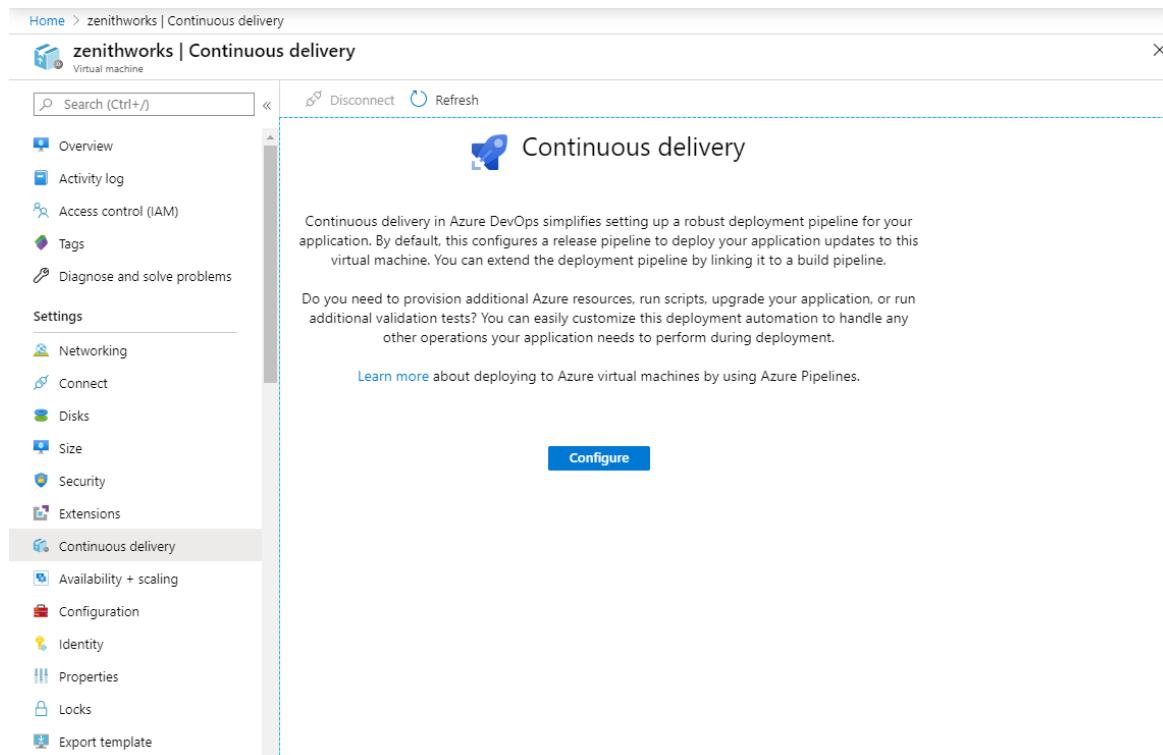
Azure Pipelines provides a fully featured set of CI/CD automation tools for deployments to virtual machines. This article will show you how to set up a classic release pipeline that uses the blue-green strategy to deploy to Linux virtual machines. Azure also supports other strategies like [rolling](#) and [canary](#) deployments.

## Blue-green deployments

A blue-green deployment is a deployment strategy where you create two separate and identical environments but only one is live at any time. This strategy is used to increase availability and reduce downtime by switching between the blue/green environments. The blue environment is usually set to run the current version of the application while the green environment is set to host the updated version. When all updates are completed, traffic is directed to the green environment and blue environment is set to idle.

Using the [Continuous-delivery](#) feature, you can use the blue-green deployment strategy to deploy to your virtual machines from Azure portal.

1. Sign in to [Azure portal](#) and navigate to a virtual machine.
2. Select [Continuous delivery](#), and then select [Configure](#).



3. In the configuration panel, select **Use existing** and select your organization/project or select **Create** and create new ones.
4. Select your **Deployment group name** from the dropdown menu or create a new one.
5. Select your **Build pipeline** from the dropdown menu.

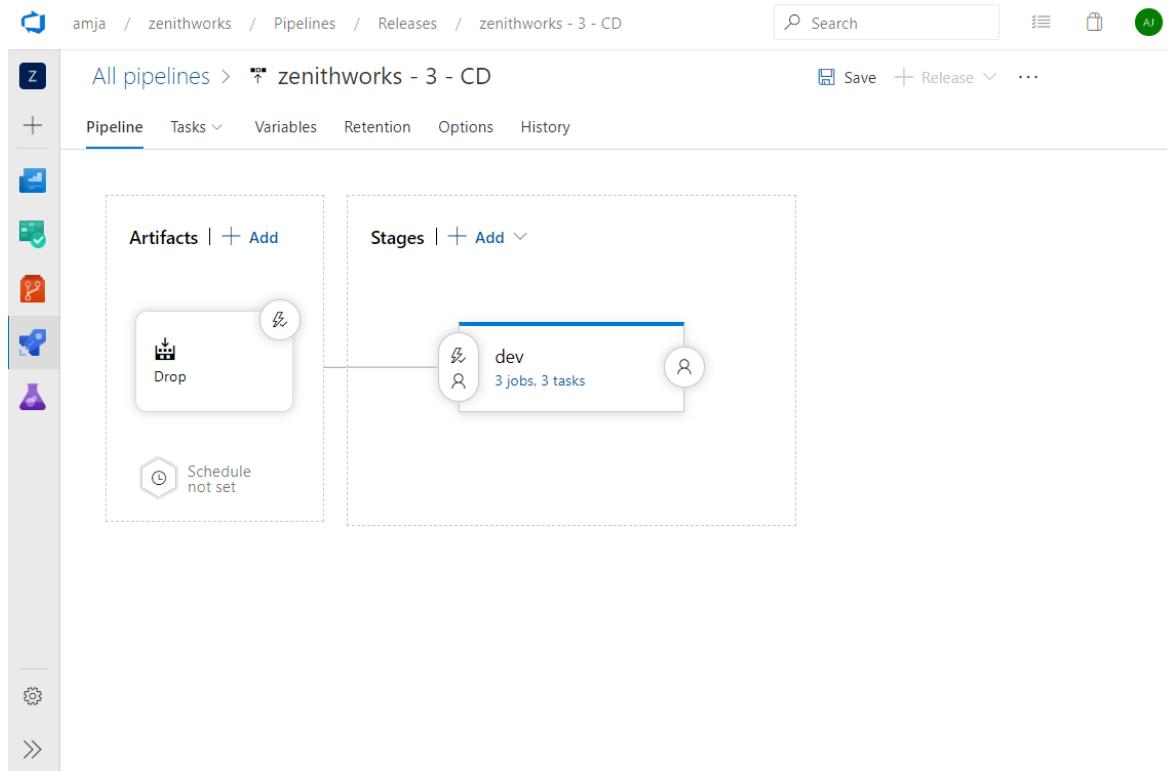
6. Select the Deployment strategy dropdown menu, and then select **Blue-Green**.

The screenshot shows the Azure portal interface for configuring continuous delivery. On the left, there's a sidebar with various settings like Overview, Activity log, and Tags. The main area is titled 'Continuous delivery' and contains instructions about setting up a release pipeline. A prominent 'Configure' button is at the bottom. To the right, a detailed configuration dialog is open. It asks for the Azure DevOps Organization (set to 'amja') and the Project ('zenithworks'). It also defines a 'Deployment group' and 'Deployment group name' ('production-dg'), and specifies a 'Build pipeline' ('zw-core-CI'). The 'Deployment strategy' dropdown is open, showing 'Rolling' and 'Blue-Green' as options, with 'Blue-Green' highlighted. At the bottom right of the dialog is a large blue 'OK' button.

7. Add a "blue" or "green" tag to VMs that are used for blue-green deployments. If a VM is for a standby role, tag it as "green". Otherwise, tag it as "blue".

This screenshot is similar to the previous one but includes a 'Tags' section at the bottom of the configuration dialog. This section contains a single tag labeled 'green'. The rest of the dialog is identical to the first screenshot, showing the organization 'amja', project 'zenithworks', deployment group 'production-dg', build pipeline 'zw-core-CI', and the 'Deployment strategy' set to 'Blue-Green'.

8. Select **OK** to configure the classic release pipeline to deploy to your virtual machine.



9. Navigate to your release pipeline and then select **Edit** to view the pipeline configuration. In this example, the *dev* stage is composed of three jobs:
  - a. Deploy Green: the app is deployed to a standby VM tagged "green".
  - b. Wait for manual resumption: the pipeline pauses and waits for manual intervention.
  - c. Swap Blue-Green: this job swaps the "blue" and "green" tags in the VMs. This ensures that VMs with older application versions are now tagged as "green". During the next pipeline run, applications will be deployed to these VMs.

The screenshot shows the Azure DevOps Pipeline configuration for the release pipeline 'zenithworks - 3 - CD'. The pipeline has two stages: 'Drop' and 'dev'. The 'dev' stage is currently selected. The 'dev' stage contains three tasks: 'Deploy Green', 'Execute Deploy Script', and 'Wait for Manual resumption'. The 'Deploy Green' task is selected. The configuration pane on the right shows the deployment group 'production-dg' selected, and the 'Required tags' field contains 'green'. Other settings include 'Targets to deploy in parallel' set to 'Multiple' and 'Maximum number of targets in parallel' set to 100%.

## Resources

- [Deploy to Azure virtual machines with Azure DevOps](#)
- [Deploy to an Azure virtual machine scale set](#)

## Related articles

- [Configure the rolling deployment strategy](#)
- [Configure the canary deployment strategy](#)

# Common Azure CLI commands for managing Azure resources

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

The Azure CLI allows you to create and manage your Azure resources on macOS, Linux, and Windows. This article details some of the most common commands to create and manage virtual machines (VMs).

This article requires the Azure CLI version 2.0.4 or later. Run `az --version` to find the version. If you need to upgrade, see [Install Azure CLI](#). You can also use [Cloud Shell](#) from your browser.

## Basic Azure Resource Manager commands in Azure CLI

For more detailed help with specific command line switches and options, you can use the online command help and options by typing `az <command> <subcommand> --help`.

### Create VMs

TASK	AZURE CLI COMMANDS
Create a resource group	<code>az group create --name myResourceGroup --location eastus</code>
Create a Linux VM	<code>az vm create --resource-group myResourceGroup --name myVM --image ubuntults</code>
Create a Windows VM	<code>az vm create --resource-group myResourceGroup --name myVM --image win2016datacenter</code>

### Manage VM state

TASK	AZURE CLI COMMANDS
Start a VM	<code>az vm start --resource-group myResourceGroup --name myVM</code>
Stop a VM	<code>az vm stop --resource-group myResourceGroup --name myVM</code>
Deallocate a VM	<code>az vm deallocate --resource-group myResourceGroup --name myVM</code>
Restart a VM	<code>az vm restart --resource-group myResourceGroup --name myVM</code>
Redeploy a VM	<code>az vm redeploy --resource-group myResourceGroup --name myVM</code>
Delete a VM	<code>az vm delete --resource-group myResourceGroup --name myVM</code>

### Get VM info

TASK	AZURE CLI COMMANDS
List VMs	<code>az vm list</code>
Get information about a VM	<code>az vm show --resource-group myResourceGroup --name myVM</code>
Get usage of VM resources	<code>az vm list-usage --location eastus</code>
Get all available VM sizes	<code>az vm list-sizes --location eastus</code>

## Disk and images

TASK	AZURE CLI COMMANDS
Add a data disk to a VM	<code>az vm disk attach --resource-group myResourceGroup --vm-name myVM --disk myDataDisk --size-gb 128 --new</code>
Remove a data disk from a VM	<code>az vm disk detach --resource-group myResourceGroup --vm-name myVM --disk myDataDisk</code>
Resize a disk	<code>az disk update --resource-group myResourceGroup --name myDataDisk --size-gb 256</code>
Snapshot a disk	<code>az snapshot create --resource-group myResourceGroup --name mySnapshot --source myDataDisk</code>
Create image of a VM	<code>az image create --resource-group myResourceGroup --source myVM --name myImage</code>
Create VM from image	<code>az vm create --resource-group myResourceGroup --name myNewVM --image myImage</code>

## Next steps

For additional examples of the CLI commands, see the [Create and Manage Linux VMs with the Azure CLI](#) tutorial.

# Common PowerShell commands for creating and managing Azure Virtual Machines

9/21/2022 • 2 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

This article covers some of the Azure PowerShell commands that you can use to create and manage virtual machines in your Azure subscription. For more detailed help with specific command-line switches and options, you can use the [Get-Help command](#).

These variables might be useful for you if running more than one of the commands in this article:

- \$location - The location of the virtual machine. You can use [Get-AzLocation](#) to find a geographical region that works for you.
- \$myResourceGroup - The name of the resource group that contains the virtual machine.
- \$myVM - The name of the virtual machine.

## Create a VM - simplified

TASK	COMMAND
Create a simple VM	<code>New-AzVM -Name \$myVM</code>  New-AzVM has a set of <i>simplified</i> parameters, where all that is required is a single name. The value for -Name will be used as the name for all of the resources required for creating a new VM. You can specify more, but this is all that is required.
Create a VM from a custom image	<code>New-AzVm -ResourceGroupName \$myResourceGroup -Name \$myVM ImageName "myImage" -Location \$location</code>  You need to have already created your own <a href="#">managed image</a> . You can use an image to make multiple, identical VMs.

## Create a VM configuration

TASK	COMMAND
Create a VM configuration	<code>\$vm = New-AzVMConfig -VMName \$myVM -VMSize "Standard_D1_v1"</code>  The VM configuration is used to define or update settings for the VM. The configuration is initialized with the name of the VM and its <a href="#">size</a> .

TASK	COMMAND
Add configuration settings	\$vm = <a href="#">Set-AzVMOperatingSystem</a> -VM \$vm -Windows -ComputerName \$myVM -Credential \$cred -ProvisionVMAgent -EnableAutoUpdate
	<p>Operating system settings including <a href="#">credentials</a> are added to the configuration object that you previously created using <a href="#">New-AzVMConfig</a>.</p>
Add a network interface	\$vm = <a href="#">Add-AzVMNetworkInterface</a> -VM \$vm -Id \$nic.Id
	<p>A VM must have a <a href="#">network interface</a> to communicate in a virtual network. You can also use <a href="#">Get-AzNetworkInterface</a> to retrieve an existing network interface object.</p>
Specify a platform image	\$vm = <a href="#">Set-AzVMSourceImage</a> -VM \$vm -PublisherName "publisher_name" -Offer "publisher_offer" -Skus "product_sku" -Version "latest"
	<p><a href="#">Image information</a> is added to the configuration object that you previously created using <a href="#">New-AzVMConfig</a>. The object returned from this command is only used when you set the OS disk to use a platform image.</p>
Create a VM	<a href="#">New-AzVM</a> -ResourceGroupName \$myResourceGroup -Location \$location -VM \$vm
	<p>All resources are created in a <a href="#">resource group</a>. Before you run this command, run <a href="#">New-AzVMConfig</a>, <a href="#">Set-AzVMOperatingSystem</a>, <a href="#">Set-AzVMSourceImage</a>, <a href="#">Add-AzVMNetworkInterface</a>, and <a href="#">Set-AzVMOSDisk</a>.</p>
Update a VM	<a href="#">Update-AzVM</a> -ResourceGroupName \$myResourceGroup -VM \$vm
	<p>Get the current VM configuration using <a href="#">Get-AzVM</a>, change configuration settings on the VM object, and then run this command.</p>

## Get information about VMs

TASK	COMMAND
List VMs in a subscription	<a href="#">Get-AzVM</a>

TASK	COMMAND
List VMs in a resource group	<code>Get-AzVM -ResourceGroupName \$myResourceGroup</code>
	To get a list of resource groups in your subscription, use <a href="#">Get-AzResourceGroup</a> .
Get information about a VM	<code>Get-AzVM -ResourceGroupName \$myResourceGroup -Name \$myVM</code>

## Manage VMs

TASK	COMMAND
Start a VM	<code>Start-AzVM -ResourceGroupName \$myResourceGroup -Name \$myVM</code>
Stop a VM	<code>Stop-AzVM -ResourceGroupName \$myResourceGroup -Name \$myVM</code>
Restart a running VM	<code>Restart-AzVM -ResourceGroupName \$myResourceGroup -Name \$myVM</code>
Delete a VM	<code>Remove-AzVM -ResourceGroupName \$myResourceGroup -Name \$myVM</code>

## Next steps

- See the basic steps for creating a virtual machine in [Create a Windows VM using Resource Manager and PowerShell](#).

# Common PowerShell commands for Azure Virtual Networks

9/21/2022 • 4 minutes to read • [Edit Online](#)

Applies to: ✓ Linux VMs ✓ Windows VMs

If you want to create a virtual machine, you need to create a [virtual network](#) or know about an existing virtual network in which the VM can be added. Typically, when you create a VM, you also need to consider creating the resources described in this article.

See [How to install and configure Azure PowerShell](#) for information about installing the latest version of Azure PowerShell, selecting your subscription, and signing in to your account.

Some variables might be useful for you if running more than one of the commands in this article:

- \$location - The location of the network resources. You can use [Get-AzLocation](#) to find a [geographical region](#) that works for you.
- \$myResourceGroup - The name of the resource group where the network resources are located.

## Create network resources

TASK	COMMAND
Create subnet configurations	\$subnet1 = <a href="#">New-AzVirtualNetworkSubnetConfig</a> -Name "mySubnet1" -AddressPrefix XX.X.X.X/XX \$subnet2 = <a href="#">New-AzVirtualNetworkSubnetConfig</a> -Name "mySubnet2" -AddressPrefix XX.X.X.X/XX  A typical network might have a subnet for an <a href="#">internet facing load balancer</a> and a separate subnet for an <a href="#">internal load balancer</a> .
Create a virtual network	\$vnet = <a href="#">New-AzVirtualNetwork</a> -Name "myVNet" -ResourceGroupName \$myResourceGroup -Location \$location -AddressPrefix XX.X.X.X/XX -Subnet \$subnet1, \$subnet2
Test for a unique domain name	<a href="#">Test-AzDnsAvailability</a> -DomainNameLabel "myDNS" -Location \$location  You can specify a DNS domain name for a <a href="#">public IP resource</a> , which creates a mapping for domainname.location.cloudapp.azure.com to the public IP address in the Azure-managed DNS servers. The name can contain only letters, numbers, and hyphens. The first and last character must be a letter or number and the domain name must be unique within its Azure location. If <b>True</b> is returned, your proposed name is globally unique.

TASK	COMMAND
Create a public IP address	<pre>\$pip = <a href="#">New-AzPublicIpAddress</a> -Name "myPublicIp" -ResourceGroupName \$myResourceGroup -DomainNameLabel "myDNS" -Location \$location -AllocationMethod Dynamic</pre> <p>The public IP address uses the domain name that you previously tested and is used by the frontend configuration of the load balancer.</p>
Create a frontend IP configuration	<pre>\$frontendIP = <a href="#">New-AzLoadBalancerFrontendIpConfig</a> -Name "myFrontendIP" -PublicIpAddress \$pip</pre> <p>The frontend configuration includes the public IP address that you previously created for incoming network traffic.</p>
Create a backend address pool	<pre>\$beAddressPool = <a href="#">New-AzLoadBalancerBackendAddressPoolConfig</a> -Name "myBackendAddressPool"</pre> <p>Provides internal addresses for the backend of the load balancer that are accessed through a network interface.</p>
Create a probe	<pre>\$healthProbe = <a href="#">New-AzLoadBalancerProbeConfig</a> -Name "myProbe" -RequestPath 'HealthProbe.aspx' -Protocol http -Port 80 -IntervalInSeconds 15 -ProbeCount 2</pre> <p>Contains health probes used to check availability of virtual machines instances in the backend address pool.</p>
Create a load balancing rule	<pre>\$lbRule = <a href="#">New-AzLoadBalancerRuleConfig</a> -Name HTTP -FrontendIpConfiguration \$frontendIP -BackendAddressPool \$beAddressPool -Probe \$healthProbe -Protocol Tcp -FrontendPort 80 -BackendPort 80</pre> <p>Contains rules that assign a public port on the load balancer to a port in the backend address pool.</p>
Create an inbound NAT rule	<pre>\$inboundNATRule = <a href="#">New-AzLoadBalancerInboundNatRuleConfig</a> -Name "myInboundRule1" -FrontendIpConfiguration \$frontendIP -Protocol TCP -FrontendPort 3441 -BackendPort 3389</pre> <p>Contains rules mapping a public port on the load balancer to a port for a specific virtual machine in the backend address pool.</p>
Create a load balancer	<pre>\$loadBalancer = <a href="#">New-AzLoadBalancer</a> -ResourceGroupName \$myResourceGroup -Name "myLoadBalancer" -Location \$location -FrontendIpConfiguration \$frontendIP -InboundNatRule \$inboundNATRule -LoadBalancingRule \$lbRule -BackendAddressPool \$beAddressPool -Probe \$healthProbe</pre>

TASK	COMMAND
Create a network interface	<pre>\$nic1 = New-AzNetworkInterface -ResourceGroupName \$myResourceGroup -Name "myNIC" -Location \$location - PrivateIpAddress XX.X.X.X -Subnet \$subnet2 - LoadBalancerBackendAddressPool \$loadBalancer.BackendAddressPools[0] - LoadBalancerInboundNatRule \$loadBalancer.InboundNatRules[0]</pre> <p>Create a network interface using the public IP address and virtual network subnet that you previously created.</p>

## Get information about network resources

TASK	COMMAND
List virtual networks	<pre>Get-AzVirtualNetwork -ResourceGroupName \$myResourceGroup</pre> <p>Lists all the virtual networks in the resource group.</p>
Get information about a virtual network	<pre>Get-AzVirtualNetwork -Name "myVNet" - ResourceGroupName \$myResourceGroup</pre>
List subnets in a virtual network	<pre>Get-AzVirtualNetwork -Name "myVNet" - ResourceGroupName \$myResourceGroup   Select Subnets</pre>
Get information about a subnet	<pre>Get-AzVirtualNetworkSubnetConfig -Name "mySubnet1" - VirtualNetwork \$vnet</pre> <p>Gets information about the subnet in the specified virtual network. The \$vnet value represents the object returned by Get-AzVirtualNetwork.</p>
List IP addresses	<pre>Get-AzPublicIpAddress -ResourceGroupName \$myResourceGroup</pre> <p>Lists the public IP addresses in the resource group.</p>
List load balancers	<pre>Get-AzLoadBalancer -ResourceGroupName \$myResourceGroup</pre> <p>Lists all the load balancers in the resource group.</p>
List network interfaces	<pre>Get-AzNetworkInterface -ResourceGroupName \$myResourceGroup</pre> <p>Lists all the network interfaces in the resource group.</p>
Get information about a network interface	<pre>Get-AzNetworkInterface -Name "myNIC" - ResourceGroupName \$myResourceGroup</pre> <p>Gets information about a specific network interface.</p>

TASK	COMMAND
Get the IP configuration of a network interface	<p><code>Get-AzNetworkInterfaceIPConfig -Name "myNICIP" -NetworkInterface \$nic</code></p> <p>Gets information about the IP configuration of the specified network interface. The \$nic value represents the object returned by <code>Get-AzNetworkInterface</code>.</p>

## Manage network resources

TASK	COMMAND
Add a subnet to a virtual network	<p><code>Add-AzVirtualNetworkSubnetConfig -AddressPrefix XX.X.X.X/XX -Name "mySubnet1" -VirtualNetwork \$vnet</code></p> <p>Adds a subnet to an existing virtual network. The \$vnet value represents the object returned by <code>Get-AzVirtualNetwork</code>.</p>
Delete a virtual network	<p><code>Remove-AzVirtualNetwork -Name "myVNet" -ResourceGroupName \$myResourceGroup</code></p> <p>Removes the specified virtual network from the resource group.</p>
Delete a network interface	<p><code>Remove-AzNetworkInterface -Name "myNIC" -ResourceGroupName \$myResourceGroup</code></p> <p>Removes the specified network interface from the resource group.</p>
Delete a load balancer	<p><code>Remove-AzLoadBalancer -Name "myLoadBalancer" -ResourceGroupName \$myResourceGroup</code></p> <p>Removes the specified load balancer from the resource group.</p>
Delete a public IP address	<p><code>Remove-AzPublicIpAddress -Name "myIPAddress" -ResourceGroupName \$myResourceGroup</code></p> <p>Removes the specified public IP address from the resource group.</p>

## Next Steps

Use the network interface that you just created when you [create a VM](#).

# Virtual machines in an Azure Resource Manager template

9/21/2022 • 10 minutes to read • [Edit Online](#)

Applies to: ✓ Windows VMs

This article describes aspects of an Azure Resource Manager template that apply to virtual machines. This article doesn't describe a complete template for creating a virtual machine; for that you need resource definitions for storage accounts, network interfaces, public IP addresses, and virtual networks. For more information about how these resources can be defined together, see the [Resource Manager template walkthrough](#).

There are many [templates in the gallery](#) that include the VM resource. Not all elements that can be included in a template are described here.

This example shows a typical resource section of a template for creating a specified number of VMs:

```
"resources": [
  {
    "apiVersion": "2016-04-30-preview",
    "type": "Microsoft.Compute/virtualMachines",
    "name": "[concat('myVM', copyindex())]",
    "location": "[resourceGroup().location]",
    "copy": {
      "name": "virtualMachineLoop",
      "count": "[parameters('numberOfInstances')]"
    },
    "dependsOn": [
      "[concat('Microsoft.Network/networkInterfaces/myNIC', copyindex())]"
    ],
    "properties": {
      "hardwareProfile": {
        "vmSize": "Standard_DS1"
      },
      "osProfile": {
        "computerName": "[concat('myVM', copyindex())]",
        "adminUsername": "[parameters('adminUsername')]",
        "adminPassword": "[parameters('adminPassword')]"
      },
      "storageProfile": {
        "imageReference": {
          "publisher": "MicrosoftWindowsServer",
          "offer": "WindowsServer",
          "sku": "2012-R2-Datacenter",
          "version": "latest"
        },
        "osDisk": {
          "name": "[concat('myOSDisk', copyindex())]",
          "caching": "ReadWrite",
          "createOption": "FromImage"
        },
        "dataDisks": [
          {
            "name": "[concat('myDataDisk', copyindex())]",
            "diskSizeGB": "100",
            "lun": 0,
            "createOption": "Empty"
          }
        ]
      }
    }
  }
]
```

```

"networkProfile": {
    "networkInterfaces": [
        {
            "id": "[resourceId('Microsoft.Network/networkInterfaces',
                concat('myNIC', copyindex()))]"
        }
    ]
},
"diagnosticsProfile": {
    "bootDiagnostics": {
        "enabled": "true",
        "storageUri": "[concat('https://', variables('storageName'), '.blob.core.windows.net')]"
    }
}
},
"resources": [
{
    "name": "Microsoft.Insights.VMDiagnosticsSettings",
    "type": "extensions",
    "location": "[resourceGroup().location]",
    "apiVersion": "2016-03-30",
    "dependsOn": [
        "[concat('Microsoft.Compute/virtualMachines/myVM', copyindex())]"
    ],
    "properties": {
        "publisher": "Microsoft.Azure.Diagnostics",
        "type": "IaaS.Diagnostics",
        "typeHandlerVersion": "1.5",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "xmlCfg": "[base64(concat(variables('wadcfgxstart'),
                variables('wadmetricsresourceid'),
                concat('myVM', copyindex()),
                variables('wadcfgxend'))))",
            "storageAccount": "[variables('storageName')]"
        },
        "protectedSettings": {
            "storageAccountName": "[variables('storageName')]",
            "storageAccountKey": "[listkeys(variables('accountid'),
                '2015-06-15').key1]",
            "storageAccountEndPoint": "https://core.windows.net"
        }
    }
},
{
    "name": "MyCustomScriptExtension",
    "type": "extensions",
    "apiVersion": "2016-03-30",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[concat('Microsoft.Compute/virtualMachines/myVM', copyindex())]"
    ],
    "properties": {
        "publisher": "Microsoft.Compute",
        "type": "CustomScriptExtension",
        "typeHandlerVersion": "1.7",
        "autoUpgradeMinorVersion": true,
        "settings": {
            "fileUris": [
                "[concat('https://', variables('storageName'),
                    '.blob.core.windows.net/customscripts/start.ps1')]"
            ],
            "commandToExecute": "powershell.exe -ExecutionPolicy Unrestricted -File start.ps1"
        }
    }
}
]
}
]
```

## NOTE

This example relies on a storage account that was previously created. You could create the storage account by deploying it from the template. The example also relies on a network interface and its dependent resources that would be defined in the template. These resources are not shown in the example.

## API Version

When you deploy resources using a template, you have to specify a version of the API to use. The example shows the virtual machine resource using this `apiVersion` element:

```
"apiVersion": "2016-04-30-preview",
```

The version of the API you specify in your template affects which properties you can define in the template. In general, you should select the most recent API version when creating templates. For existing templates, you can decide whether you want to continue using an earlier API version, or update your template for the latest version to take advantage of new features.

Use these opportunities for getting the latest API versions:

- REST API - [List all resource providers](#)
- PowerShell - [Get-AzResourceProvider](#)
- Azure CLI - [az provider show](#)

## Parameters and variables

**Parameters** make it easy for you to specify values for the template when you run it. This parameters section is used in the example:

```
"parameters": {  
    "adminUsername": { "type": "string" },  
    "adminPassword": { "type": "securestring" },  
    "numberOfInstances": { "type": "int" }  
},
```

When you deploy the example template, you enter values for the name and password of the administrator account on each VM and the number of VMs to create. You have the option of specifying parameter values in a separate file that's managed with the template, or providing values when prompted.

**Variables** make it easy for you to set up values in the template that are used repeatedly throughout it or that can change over time. This variables section is used in the example:

```

"variables": {
    "storageName": "mystore1",
    "accountid": "[concat('/subscriptions/', subscription().subscriptionId,
        '/resourceGroups/', resourceGroup().name,
        '/providers/', 'Microsoft.Storage/storageAccounts/', variables('storageName'))]",
    "wadlogs": "<WadCfg>
        <DiagnosticMonitorConfiguration overallQuotaInMB=\"4096\" xmlns=\"http://schemas.microsoft.com/ServiceHosting/2010/10/DiagnosticsConfiguration\">
            <DiagnosticInfrastructureLogs scheduledTransferLogLevelFilter=\"Error\"/>
            <WindowsEventLog scheduledTransferPeriod=\"PT1M\" >
                <DataSource name=\"Application!*[System[(Level = 1 or Level = 2)]]\" />
                <DataSource name=\"Security!*[System[(Level = 1 or Level = 2)]]\" />
                <DataSource name=\"System!*[System[(Level = 1 or Level = 2)]]\" />
            </WindowsEventLog>",
        <wadperfcounters": "<PerformanceCounters scheduledTransferPeriod=\"PT1M\">
            <PerformanceCounterConfiguration counterSpecifier=\"\\Process(_Total)\\Thread Count\" sampleRate=\"PT15S\" unit=\"Count\">
                <annotation displayName=\"Threads\" locale=\"en-us\"/>
            </PerformanceCounterConfiguration>
        </PerformanceCounters>",
        "wadcfgxstart": "[concat(variables('wadlogs'), variables('wadperfcounters'),
            '<Metrics resourceId=\"\"\'])",
        "wadmetricsresourceid": "[concat('/subscriptions/', subscription().subscriptionId,
            '/resourceGroups/', resourceGroup().name ,
            '/providers/', 'Microsoft.Compute/virtualMachines/')]",
        "wadcfgxend": "\"><MetricAggregation scheduledTransferPeriod=\"PT1H\"/>
            <MetricAggregation scheduledTransferPeriod=\"PT1M\"/>
        </Metrics></DiagnosticMonitorConfiguration>
    </WadCfg>"}
},

```

When you deploy the example template, variable values are used for the name and identifier of the previously created storage account. Variables are also used to provide the settings for the diagnostic extension. Use the [best practices for creating Azure Resource Manager templates](#) to help you decide how you want to structure the parameters and variables in your template.

## Resource loops

When you need more than one virtual machine for your application, you can use a copy element in a template. This optional element loops through creating the number of VMs that you specified as a parameter:

```

"copy": {
    "name": "virtualMachineLoop",
    "count": "[parameters('numberOfInstances')]"
},

```

Also, notice in the example that the loop index is used when specifying some of the values for the resource. For example, if you entered an instance count of three, the names of the operating system disks are myOSDisk1, myOSDisk2, and myOSDisk3:

```

"osDisk": {
    "name": "[concat('myOSDisk', copyindex())]",
    "caching": "ReadWrite",
    "createOption": "FromImage"
}

```

#### NOTE

This example uses managed disks for the virtual machines.

Keep in mind that creating a loop for one resource in the template may require you to use the loop when creating or accessing other resources. For example, multiple VMs can't use the same network interface, so if your template loops through creating three VMs it must also loop through creating three network interfaces. When assigning a network interface to a VM, the loop index is used to identify it:

```
"networkInterfaces": [ {  
    "id": "[resourceId('Microsoft.Network/networkInterfaces',  
        concat('myNIC', copyindex()))]"  
} ]
```

## Dependencies

Most resources depend on other resources to work correctly. Virtual machines must be associated with a virtual network and to do that it needs a network interface. The `dependsOn` element is used to make sure that the network interface is ready to be used before the VMs are created:

```
"dependsOn": [  
    "[concat('Microsoft.Network/networkInterfaces/', 'myNIC', copyindex())]"  
,
```

Resource Manager deploys in parallel any resources that aren't dependent on another resource being deployed. Be careful when setting dependencies because you can inadvertently slow your deployment by specifying unnecessary dependencies. Dependencies can chain through multiple resources. For example, the network interface depends on the public IP address and virtual network resources.

How do you know if a dependency is required? Look at the values you set in the template. If an element in the virtual machine resource definition points to another resource that is deployed in the same template, you need a dependency. For example, your example virtual machine defines a network profile:

```
"networkProfile": {  
    "networkInterfaces": [ {  
        "id": "[resourceId('Microsoft.Network/networkInterfaces',  
            concat('myNIC', copyindex()))]"  
    } ]  
},
```

To set this property, the network interface must exist. Therefore, you need a dependency. You also need to set a dependency when one resource (a child) is defined within another resource (a parent). For example, the diagnostic settings and custom script extensions are both defined as child resources of the virtual machine. They can't be created until the virtual machine exists. Therefore, both resources are marked as dependent on the virtual machine.

## Profiles

Several profile elements are used when defining a virtual machine resource. Some are required and some are optional. For example, the `hardwareProfile`, `osProfile`, `storageProfile`, and `networkProfile` elements are required, but the `diagnosticsProfile` is optional. These profiles define settings such as:

- `size`

- [name](#) and credentials
- disk and [operating system settings](#)
- [network interface](#)
- boot diagnostics

## Disks and images

In Azure, vhd files can represent [disks or images](#). When the operating system in a vhd file is specialized to be a specific VM, it's referred to as a disk. When the operating system in a vhd file is generalized to be used to create many VMs, it's referred to as an image.

### Create new virtual machines and new disks from a platform image

When you create a VM, you must decide what operating system to use. The `imageReference` element is used to define the operating system of a new VM. The example shows a definition for a Windows Server operating system:

```
"imageReference": {
  "publisher": "MicrosoftWindowsServer",
  "offer": "WindowsServer",
  "sku": "2012-R2-Datacenter",
  "version": "latest"
},
```

If you want to create a Linux operating system, you might use this definition:

```
"imageReference": {
  "publisher": "Canonical",
  "offer": "UbuntuServer",
  "sku": "14.04.2-LTS",
  "version": "latest"
},
```

Configuration settings for the operating system disk are assigned with the `osDisk` element. The example defines a new managed disk with the caching mode set to **ReadWrite** and that the disk is being created from a [platform image](#):

```
"osDisk": {
  "name": "[concat('myOSDisk', copyindex())]",
  "caching": "ReadWrite",
  "createOption": "FromImage"
},
```

### Create new virtual machines from existing managed disks

If you want to create virtual machines from existing disks, remove the `imageReference` and the `osProfile` elements and define these disk settings:

```
"osDisk": {
  "osType": "Windows",
  "managedDisk": {
    "id": "[resourceId('Microsoft.Compute/disks', [concat('myOSDisk', copyindex())])]"
  },
  "caching": "ReadWrite",
  "createOption": "Attach"
},
```

## Create new virtual machines from a managed image

If you want to create a virtual machine from a managed image, change the `imageReference` element and define these disk settings:

```
"storageProfile": {
  "imageReference": {
    "id": "[resourceId('Microsoft.Compute/images', 'myImage')]"
  },
  "osDisk": {
    "name": "[concat('myOSDisk', copyindex())]",
    "osType": "Windows",
    "caching": "ReadWrite",
    "createOption": "FromImage"
  }
},
```

## Attach data disks

You can optionally add data disks to the VMs. The [number of disks](#) depends on the size of operating system disk that you use. With the size of the VMs set to Standard\_DS1\_v2, the maximum number of data disks that could be added to them is two. In the example, one managed data disk is being added to each VM:

```
"dataDisks": [
  {
    "name": "[concat('myDataDisk', copyindex())]",
    "diskSizeGB": "100",
    "lun": 0,
    "caching": "ReadWrite",
    "createOption": "Empty"
  }
],
```

## Extensions

Although [extensions](#) are a separate resource, they're closely tied to VMs. Extensions can be added as a child resource of the VM or as a separate resource. The example shows the [Diagnostics Extension](#) being added to the VMs:

```
{
  "name": "Microsoft.Insights.VMDiagnosticsSettings",
  "type": "extensions",
  "location": "[resourceGroup().location]",
  "apiVersion": "2016-03-30",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/myVM', copyindex())]"
  ],
  "properties": {
    "publisher": "Microsoft.Azure.Diagnostics",
    "type": "IaaSDiagnostics",
    "typeHandlerVersion": "1.5",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "xmlCfg": "[base64(concat(variables('wadcfgxstart'),
variables('wadmetricsresourceid'),
concat('myVM', copyindex()),
variables('wadcfgxend')))]",
      "storageAccount": "[variables('storageName')]"
    },
    "protectedSettings": {
      "storageAccountName": "[variables('storageName')]",
      "storageAccountKey": "[listkeys(variables('accountid'),
'2015-06-15').key1]",
      "storageAccountEndPoint": "https://core.windows.net"
    }
  }
},
}
```

This extension resource uses the storageName variable and the diagnostic variables to provide values. If you want to change the data that is collected by this extension, you can add more performance counters to the wadperfcounters variable. You could also choose to put the diagnostics data into a different storage account than where the VM disks are stored.

There are many extensions that you can install on a VM, but the most useful is probably the [Custom Script Extension](#). In the example, a PowerShell script named start.ps1 runs on each VM when it first starts:

```
{
  "name": "MyCustomScriptExtension",
  "type": "extensions",
  "apiVersion": "2016-03-30",
  "location": "[resourceGroup().location]",
  "dependsOn": [
    "[concat('Microsoft.Compute/virtualMachines/myVM', copyindex())]"
  ],
  "properties": {
    "publisher": "Microsoft.Compute",
    "type": "CustomScriptExtension",
    "typeHandlerVersion": "1.7",
    "autoUpgradeMinorVersion": true,
    "settings": {
      "fileUris": [
        "[concat('https://', variables('storageName'),
'.blob.core.windows.net/customscripts/start.ps1')]"
      ],
      "commandToExecute": "powershell.exe -ExecutionPolicy Unrestricted -File start.ps1"
    }
  }
}
```

The start.ps1 script can accomplish many configuration tasks. For example, the data disks that are added to the VMs in the example aren't initialized; you can use a custom script to initialize them. If you have multiple startup tasks to do, you can use the start.ps1 file to call other PowerShell scripts in Azure storage. The example uses

PowerShell, but you can use any scripting method that is available on the operating system that you're using.

You can see the status of the installed extensions from the Extensions settings in the portal:

NAME	TYPE	V...	STATUS
Microsoft.Insights.VMDiagnosticsSettings	Microsoft.Azur...	1.*	Provisioning succ...
MyCustomScriptExtension	Microsoft.Com...	1.*	Provisioning succ...

You can also get extension information by using the **Get-AzVMExtension** PowerShell command, the **vm extension get** Azure CLI command, or the **Get extension information** REST API.

## Deployments

When you deploy a template, Azure tracks the resources that you deployed as a group and automatically assigns a name to this deployed group. The name of the deployment is the same as the name of the template.

If you're curious about the status of resources in the deployment, view the resource group in the Azure portal:

Essentials ^

Subscription name [REDACTED]

Last deployment **7/25/2016 (Succeeded)**

Subscription ID [REDACTED]

Location Central US

It's not a problem to use the same template to create resources or to update existing resources. When you use commands to deploy templates, you have the opportunity to say which **mode** you want to use. The mode can be set to either **Complete** or **Incremental**. The default is to do incremental updates. Be careful when using the **Complete** mode because you may accidentally delete resources. When you set the mode to **Complete**, Resource Manager deletes any resources in the resource group that aren't in the template.

## Next Steps

- Create your own template using [Authoring Azure Resource Manager templates](#).
- Deploy the template that you created using [Create a Windows virtual machine with a Resource Manager template](#).
- Learn how to manage the VMs that you created by reviewing [Create and manage Windows VMs with the Azure PowerShell module](#).
- For the JSON syntax and properties of resource types in templates, see [Azure Resource Manager template reference](#).





# Support and troubleshooting for Azure VMs

9/21/2022 • 2 minutes to read • [Edit Online](#)

Here are suggestions for where you can get help when developing your Azure Virtual Machines solutions.

## Self help troubleshooting



Various articles explain how to determine, diagnose, and fix issues that you might encounter when using Azure Virtual Machines. Use these articles to troubleshoot deployment failures, unexpected restarts, connection issues and more.

For a full list of self help troubleshooting content, see [Azure Virtual Machine troubleshooting documentation](#)

## Post a question on Microsoft Q&A



For quick and reliable answers on your technical product questions from Microsoft Engineers, Azure Most Valuable Professionals (MVPs), or our expert community, engage with us on [Microsoft Q&A](#), Azure's preferred destination for community support.

If you can't find an answer to your problem using search, submit a new question to Microsoft Q&A. Use one of the following tags when asking your question:

AREA	TAG
Azure Virtual Machines	<a href="#">azure-virtual-machines</a>
Azure SQL Virtual Machines	<a href="#">azure-sql-virtual-machines</a>
Azure Virtual Machine backup	<a href="#">azure-virtual-machine-backup</a>
Azure Virtual Machine extension	<a href="#">azure-virtual-machine-extension</a>
Azure Virtual Machine Images	<a href="#">azure-virtual-machine-images</a>
Azure Virtual Machine migration	<a href="#">azure-virtual-machine-migration</a>
Azure Virtual Machine monitoring	<a href="#">azure-virtual-machine-monitoring</a>
Azure Virtual Machine networking	<a href="#">azure-virtual-machine-networking</a>
Azure Virtual Machine storage	<a href="#">azure-virtual-machine-storage</a>

AREA	TAG
Azure Virtual Machine Scale Sets	azure-virtual-machine-scale-set

## Create an Azure support request



Explore the range of [Azure support options and choose the plan](#) that best fits, whether you're a developer just starting your cloud journey or a large organization deploying business-critical, strategic applications. Azure customers can create and manage support requests in the Azure portal.

- If you already have an Azure Support Plan, [open a support request here](#).
- To sign up for a new Azure Support Plan, [compare support plans](#) and select the plan that works for you.

## Create a GitHub issue



If you need help with the language and tools used to develop and manage Azure Virtual Machines, open an issue in its repository on GitHub.

LIBRARY	GITHUB ISSUES URL
Azure PowerShell	<a href="https://github.com/Azure/azure-powershell/issues">https://github.com/Azure/azure-powershell/issues</a>
Azure CLI	<a href="https://github.com/Azure/azure-cli/issues">https://github.com/Azure/azure-cli/issues</a>
Azure REST API	<a href="https://github.com/Azure/azure-rest-api-specs/issues">https://github.com/Azure/azure-rest-api-specs/issues</a>
Azure SDK for Java	<a href="https://github.com/Azure/azure-sdk-for-java/issues">https://github.com/Azure/azure-sdk-for-java/issues</a>
Azure SDK for Python	<a href="https://github.com/Azure/azure-sdk-for-python/issues">https://github.com/Azure/azure-sdk-for-python/issues</a>
Azure SDK for .NET	<a href="https://github.com/Azure/azure-sdk-for-net/issues">https://github.com/Azure/azure-sdk-for-net/issues</a>
Azure SDK for JavaScript	<a href="https://github.com/Azure/azure-sdk-for-js/issues">https://github.com/Azure/azure-sdk-for-js/issues</a>
Jenkins	<a href="https://github.com/Azure/jenkins/issues">https://github.com/Azure/jenkins/issues</a>
Terraform	<a href="https://github.com/Azure/terraform/issues">https://github.com/Azure/terraform/issues</a>
Ansible	<a href="https://github.com/Azure/Ansible/issues">https://github.com/Azure/Ansible/issues</a>

## Stay informed of updates and new releases



Learn about important product updates, roadmap, and announcements in [Azure Updates](#).

News and information about Azure Virtual Machines is shared at the [Azure blog](#).

## Next steps

Learn more about [Azure Virtual Machines](#)